DOCUMENT RESUME

ED 385 545 TM 023 961

AUTHOR Sebrechts, Marc M.; And Others
TITLE A Research Platform for Interactive Performance Assessment in Graduate Education. GRE Board Professional Report No. 90-01P.
INSTITUTION Educational Testing Service, Princeton, N.J.
SPONS AGENCY Graduate Record Examinations Board, Princeton, N.J.
REPORT NO ETS-RR-93-08
PUB DATE Apr 93
NOTE 91p.
PUB TYPE Reports - Evaluative/Feasibility (142)

EDRS PRICE MF01/PC04 Plus Postage.
DESCRIPTORS *Algebra; College Entrance Examinations; *Computer Assisted Testing; Computer Interfaces; Constructed Response; Data Analysis; Data Collection; Educational Assessment; Educational Research; Expert Systems; *Graduate Study; Higher Education; *Interaction; Item Banks; *Scoring; Word Problems (Mathematics)
IDENTIFIERS *GIDE Computer Program; Graduate Record Examinations; *Performance Based Evaluation

ABSTRACT

This report describes the development of a new tool for assessment research in graduate education. The tool, the Algebra Assessment System, is based on GIDE, a pre-existing program that diagnostically analyzes complex constructed responses to algebra word problems. The project had three goals. The first goal was to build a generically usable interface for presenting performance tasks and collecting answers in machine-readable form. This program makes the data gathering and analysis process more efficient and permits examinees to respond in the computerized environment in which Graduate Record Examinations Program offerings will be delivered. Second, the range of problems scorable by GIDE was increased to demonstrate the feasibility of automatically analyzing complex constructed responses across a broadened domain. Finally, minor faults in GIDE's analysis mechanisms were corrected. Together, these improvements constitute a tool that can be used in future research to permit stronger inferences to operational computer-based testing environments and as the basis for an eventual interactive performance assessment prototype for graduate education. Three appendixes present the item pool, the scoring rubric, and canonical solutions and bug definitions for three problem classes. (Contains 4 tables, 3 figures, and 25 references.) (Author/SLD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 385 545

GRE
RESEARCH

A Research Platform for Interactive Performance Assessment in Graduate Education

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☑ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Marc M. Sebrechts
Randy E. Bennett
Irvin R. Katz

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
H. I. BRAUN
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

April 1993

GRE Board Professional Report No. 90-01P
ETS Research Report 93-08

ETS

Educational Testing Service, Princeton, New Jersey

BEST COPY AVAILABLE

TM023961

A Research Platform for Interactive Performance Assessment in Graduate Education

Marc M. Sebrechts
Randy E. Bennett
Irvin R. Katz

GRE Board Report No. 90-01P

April 1993

This report presents the findings of a research project funded by and carried out under the auspices of the Graduate Record Examinations Board.

Educational Testing Service, Princeton, N.J. 08541

The Graduate Record Examinations Board and Educational Testing Service are dedicated to the principle of equal opportunity, and their programs, services, and employment policies are guided by that principle.

EDUCATIONAL TESTING SERVICE, ETS, the ETS logo, GRADUATE RECORD EXAMINATIONS, and GRE are registered trademarks of Educational Testing Service.

Copyright © 1993 by Educational Testing Service. All rights reserved.

Abstract

This report describes development of a new tool for assessment research in graduate education. The tool, the Algebra Assessment System, is based on GIDE, a pre-existing program that diagnostically analyzes complex constructed responses to algebra word problems. The project had three goals. The first goal was to build a generically usable interface for presenting performance tasks and collecting answers in machine-readable form. This program makes the data gathering and analysis process more efficient and permits examinees to respond in the computerized environment in which GRE Program offerings will be delivered. Second, the range of problems scorable by GIDE was increased to demonstrate the feasibility of automatically analyzing complex constructed responses across a broadened domain. Finally, minor faults in GIDE's analysis mechanism were corrected. Together, these improvements constitute a tool that can be used (1) in future research to permit stronger inferences to operational computer-based testing environments and (2) as the basis for an eventual interactive performance assessment prototype for graduate education.

Educators in general and measurement specialists in particular are increasingly questioning the congruence of traditional tests with current conceptions of cognitive ability and domain proficiency (Mislevy, in press; Resnick & Resnick, 1990; Shepard, 1991). Among other things, concerns have been expressed about the decontextualized, well-structured nature of the tasks that compose tests; the emphasis on product; and the failure to account for qualitative aspects of the problem solution.

We have been developing tools for studying the application of cognitive principles to assessment. These tools take the form of interactive environments because the implications of cognitive science and practical exigencies will require increasing dependence on computer-based test delivery (though we recognize that for some purposes other assessment modes, including paper-and-pencil, might still be preferred).

Our tools emanate from an integrated research program on interactive performance assessment (Bennett, 1992, in press). The effort's goal is to integrate research on complex constructed-response testing, knowledge-based systems, and model-based measurement to produce cognitively and psychometrically grounded, automatic assessments. The program is being carried out in three domains--computer science, algebra, and verbal reasoning--with funding from the Graduate Record Examinations (GRE) Board, the College Board, the Richard Lounsbery Foundation, and Educational Testing Service (ETS).

The emphasis in each of these domains is somewhat different. In computer science, the focus is on contextualized problem solving in a programming environment. Tasks are presented and solved with a programming editor and compiler, so that the program can be iteratively run and modified. The solution is then automatically processed to produce a partial-credit score and a qualitative analysis. A prototype system that presents tasks, accepts responses, and provides real-time analyses now exists (Bennett, Sack, & Soloway, 1991).

In verbal reasoning, a system for presenting and scoring formulating-hypotheses problems (Frederiksen & Ward, 1978) is being developed. These ill-structured verbal problems present a realistic situation and ask the examinee to pose as many plausible causes as possible. Scoring is done by a natural-language pattern matcher (Kaplan, 1992).

The work in algebra problem solving has centered on the automatic analysis of solution processes. These process analyses are the basis for assigning numerical scores and making qualitative statements about the solution. The focus on the solution process permits inferences about important problem-solving characteristics that have been traditionally ignored in standardized testing, including planning, solution monitoring, and automaticity.

The process analyses are performed by GIDE-Algebra, a batch-processing laboratory tool (Sebrechts, LaClaire, Schooler, & Soloway, 1986). Four projects using GIDE have been undertaken with funding from the GRE Board. The first project contained both development and research components. Under the former component, 12 questions were produced based on algebra word problems from the General Test's quantitative section. Three items were generated in

each of four formats: open-ended, goal specification, equation setup, and faulty solution. Knowledge bases for GIDE to use in analyzing responses were also constructed. The research component involved two major analyses, both performed on the partial-credit scores awarded by GIDE to examinee responses. The first analysis centered on the extent to which GIDE's scores agreed with those of human raters (Sebrechts, Bennett, & Rock, 1991). Correlations between GIDE and the raters ranged from .74 to .97, with a median of .88. The second analysis focused on the structural relationship between GIDE's scores and other established indicators including, GRE quantitative scores (Bennett, Sebrechts, & Rock, 1991). This confirmatory factor analysis fitted a solution comprised of two highly correlated dimensions--GRE quantitative and constructed response (the latter composed of the four question formats enumerated above). In concert, these results suggest that GIDE can replicate the judgments of content experts reasonably well in numerically scoring solutions to this class of constructed-response algebra items, that these scores are highly related to a well-established quantitative ability measure, and that the scores signify the same general proficiency across constructed-response item formats.

The second project explored the utility of two new measurement models for qualitatively aggregating item responses and, secondarily, of theoretical notions about the cognitive structures associated with solving algebra word problems (Bennett, Sebrechts, & Yamamoto, 1991). Analyses considered both the kinds of errors examinees made and how they performed on constructed-response formats intended to differ in the cognitive operations required. The results suggested only faint consistencies in the response patterns of examinees that could be interpreted in terms of cognitive structure. The weakness of the patterns was attributed to (1) limitations of the cognitive models underlying the analysis, (2) the restricted nature of the examinee sample, and (3) the small number of items.

A third study is now underway. This project centers on evaluating agreement between GIDE and human experts in characterizing examinees' problem-solving errors. The accuracy of these qualitative characterizations is important for several reasons. First, these characterizations are the basis for GIDE's partial credit scores: points are deducted depending upon the errors discovered in the solution process. Second, the qualitative analyses might be used at the conclusion of testing to let examinees know how they did on specific problems. Finally, despite our previous results, item-level analyses should have value as a building block for more general qualitative inferences about how examinees tend to perform across items (e.g., about what strategies an individual tends to use). These inferences might be helpful in lending greater meaning to test scores, as well as in suggesting how a given examinee differs from a proficient performer.

The present report describes the fourth project, which centered on developing additional capabilities for interactive performance assessment in algebra problem solving. An interactive data collection interface was constructed for GIDE, new items and knowledge bases were added to extend the range of algebra word problems that can be automatically analyzed, and GIDE's inference mechanism was improved. Together these components constitute the Algebra Assessment System, a new tool for assessment research in graduate education.

New Capabilities

Data Collection Interface

GIDE was developed as a laboratory tool for studying issues of knowledge representation, instructional diagnosis, and problem solving. As such, it was designed as a noninteractive batch processing system. Because the work involved in creating a data collection interface for complex constructed-response items is substantial, we chose to collect data for our initial GIDE projects in paper-and-pencil form. This tack raised difficulties revolving around the transcription of responses to machine-readable form. Aside from the cost of transcribing several thousand responses, we had to develop procedures and run subsidiary analyses to ensure that the transcriptions were completed consistently and accurately. Whereas this method served the purposes of the initial studies, it was not satisfactory for continued work, not only because of the recurring costs it imposes but, more importantly, because it is one step removed from the interactive environment in which GRE Program offerings will be delivered. From this perspective, an interactive data-collection program constitutes a useful research platform and a step toward an interactive performance assessment system.

Related work from intelligent tutoring. Several interfaces for collecting algebra productions have been developed. These interfaces were created as components of intelligent tutors (Wenger, 1987). Consequently, they tend to impose a structure and contain an array of functions aimed at facilitating the development of problem-solving skill. For example, the interface developed by Singley, Anderson, and Gevins (1990) structures problem solving by identifying goals with operators, where operators are used to do such things as add numbers, solve equations, collect constants, collect like terms, move variables to one side, and move constants to the other side. Students solve problems in the system by selecting these operators from a "goalpad" and indicating operator arguments from the keypad. As the problem solution is entered, the interface shows the operators and their arguments as a mixture of text descriptions and numerals, thus highlighting what operations have been performed. The hierarchical organization of the solution is also emphasized by presenting each operator followed by the suboperators used to implement it. Feedback is provided immediately when the student makes an error.

Brown (1985) describes a similar interface developed as part of Algebraland. The student is presented with an equation and a menu of transformations that can be performed on it (e.g., cancel, commute, distribute). Transformations are selected and instantiated on the equation or any of its subexpressions. Once instantiated, the transformation can be executed to produce a new expression that will, it is hoped, lead the student closer to the problem solution. Thus, the system performs all the algebraic operations, leaving for the student decisions about what operator to apply and where to apply it. Solutions are represented in two windows. The "search space" window provides a trace of the student's solution path in the form of a tree diagram, including false starts and wrong paths. The record window contains the transformations and intermediate expressions that led to the correct solution.

The algebra tutor built by McArthur, Stasz, and Hotta (1986-87) allows equations to be entered directly using the keyboard or by writing on an electronic tablet. The tablet recognizes characters and passes them to the interface, which displays them in a "WorkSpace." As each line is completed, it appears as a tree structure in a second window, the "DisplaySpace," with the current step indicated by a boxed expression. Various forms of problem-solving assistance can be accessed by clicking on a menu item and selecting a problem-solving step. Options include verifying the validity of a step, viewing the intermediate transformations leading from one step to another, generating the next step, and getting an explanation for a step.

Conceptual requirements for an algebra assessment interface. The intelligent tutoring work offers a basis from which to establish conceptual guidelines for our interface. However, because we were developing an assessment--as opposed to instructional--system, some of the conceptual requirements differ from those of intelligent tutoring.

The guidelines underlying the Algebra Assessment System interface were:

1. Ease of learning. In thinking about interface design it is useful to distinguish two software applications classes. One class subsumes those programs--such as word processors, spreadsheets, and instructional systems--used heavily by an individual. Investing time in learning is tolerable because this start-up period constitutes only a small fraction of the total time the system will be used. The second class, into which most testing applications fall, involves those programs that are used very lightly. For such applications, a lengthy up-front investment is not tolerable.

2. Ease of use. In addition to being easy to learn, the system should be simple to negotiate once it is learned. Entering and editing responses should be straightforward. Lack of keyboard facility should not impede the examinee.

3. Problem-solving fidelity. Because we are primarily interested in assessing rather than engendering problem-solving skill, we wish to observe that proficiency under conditions that are as naturalistic as possible. In paper-and-pencil form, the constraints on problem solving tend to be minimal: Examinees are free to write anywhere on the page in any direction, draw diagrams, formulate syntactically incorrect equations, and do calculations mentally.

4. Response interpretability. Whereas loose constraints characterize naturalistic problem solving, they can pose considerable problems for automatic analysis. Thus, some constraints need to be imposed to facilitate machine understanding. The challenge is in finding an appropriate balance that maximally facilitates automatic analysis and does minimal damage to problem-solving fidelity.

5. Experimental utility. One of the benefits of computer-based delivery is the ability to vary constraints experimentally. As a research tool, the data collection interface should permit constraints of interest to be switched on and off by the experimenter, thereby providing a tool for studying the effects of constraints on solution interpretability, psychometric characteristics, and problem solving.

6. Extensibility. Our data-collection interface should be a building block toward an interactive performance assessment system. As such, the interface must be designed and written to be easily extensible. For example, the interface should be capable of being simply and seamlessly linked to a response analyzer, such as GIDE, so that feedback can be provided to the examinee at the conclusion of the testing session or, alternatively, as the solutions to individual items are completed. Links should also be easily made to measurement modeling routines that may eventually be added to permit the aggregation of performance across item responses. Finally, the interface should be capable of incorporating mechanisms for nonalgebraic modes of problem representation.

7. Portability. GIDE currently runs on a Macintosh computer. The ETS computer-based test centers, however, support only MS-DOS compatible machines. The interface should be written in a language that will permit it to be ported with as little effort as possible between the Macintosh and MS-DOS environments.

8. General consistency with OSA standards. The ETS computer-based test centers support software written in conformance with the corporation's Open Systems Architecture (OSA) standards. Among other things, these standards define conventions for machine-user interaction and screen display. The screen display standards provide for a title bar with test information at the top, item presentation in the middle region below the title bar, and a set of primary icons at the bottom that control various test-level functions (e.g., help). General conformance with OSA standards makes it easier to implement a response type within the OSA software environment. Software developed to work within the OSA environment can not only more easily take advantage of the network of ETS test centers, but also take advantage of OSA's extensive software services (e.g., data management, error handling).

Interface description. Based on the above guidelines, a preliminary data collection interface was designed. This interface was pilot tested with 10 Catholic University students who had recently taken the GRE General Test. Project staff observed each testing session and interviewed each examinee. The initial interface was then revised.

A representation of the revised interface is depicted in Figure 1. The interface presents an algebra word problem, permits the examinee to enter a step-by-step solution, and records the response. The interface has not been merged with GIDE and, consequently, does not yet perform real-time analysis or provide immediate examinee feedback.

The interface is structured to conform generally to OSA standards. At the top of the screen, a title bar indicates the problem number and time remaining. Below the bar, the problem stem appears in the left window and format information to the right (e.g., the givens and goals, general equations for solution, or a faulty response to the problem). The format information cannot, in this version, be operated upon (e.g., a faulty solution cannot be edited). We may wish in future versions to move the format information into the response area where it can be manipulated.

A scrollable workspace is in the center of the screen. Students can enter any equation consisting of whole numbers, decimals, and fractions

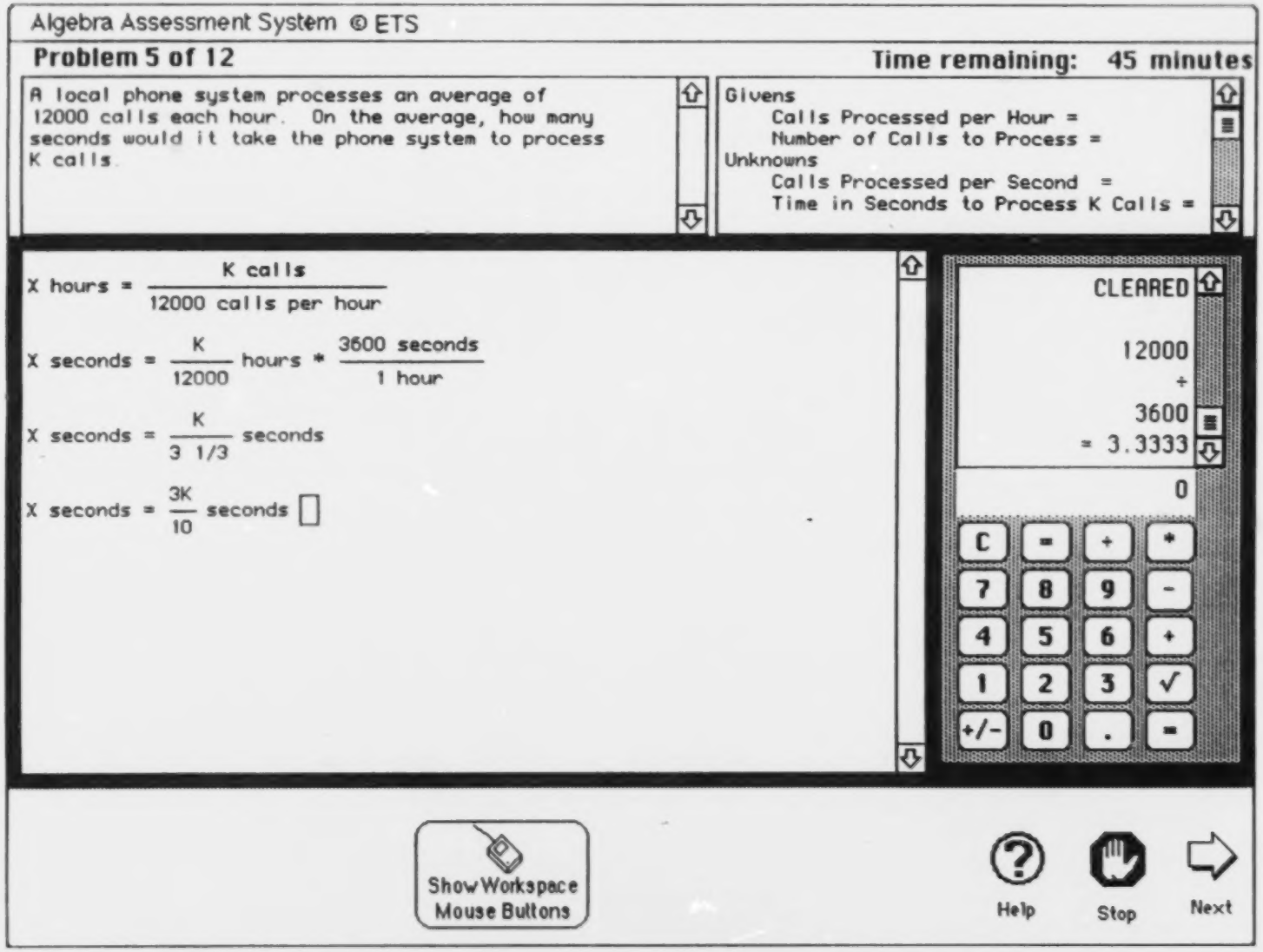
Algebra Assessment System © ETS
Problem 5 of 12
Time remaining: 45 minutes
A local phone system processes an average of 12000 calls each hour. On the average, how many seconds would it take the phone system to process K calls.
Givens
Calls Processed per Hour =
Number of Calls to Process =
Unknowns
Calls Processed per Second =
Time in Seconds to Process K Calls =
X hours = K calls / 12000 calls per hour
X seconds = K/12000 hours * 3600 seconds / 1 hour
X seconds = K/3 1/3 seconds
X seconds = 3K/10 seconds
CLEARED
12000
÷
3600
= 3.3333
0
C = + * 7 8 9 - 4 5 6 + 1 2 3 ✓ +/- 0 . =
Show Workspace Mouse Buttons
Help
Stop
Next

Figure 1. The Algebra Assessment System Interface.

together with alphanumeric labels. Equations are entered using an equation generator either by typing on the keyboard or by clicking the mouse on screen "buttons" located below the workspace. The generator's "keys" are arranged to reflect the layout of the computer keyboard and include the numerals 0-9, letters for use as variable and constant names, the four primary mathematical operators, and parentheses.

The examinee can position the cursor anywhere in the workspace using the mouse or the cursor-control keys. To make the interface easy to learn, editing functions are limited to inserting and deleting individual characters, a reasonable restriction for mathematical equations of the length required to solve algebra word problems. Insert capability is always enabled, so that once the cursor is positioned, new characters can immediately be placed between existing ones.

When entering or editing, the interface highlights the current conceptual unit within an equation. This facility is intended to reduce ambiguity in the response by helping the examinee understand how the system will interpret each equation. The current conceptual unit, or "active entity," is represented as an expanding box, the right-hand edge of which flashes. The active entity consists of whatever would be the argument to a subsequent operator. Examples are:

rate = 285 miles

and

rate = 285 x miles.

The box thus attempts to call attention to the fact that the right-hand side of the top expression will be interpreted as a single unit composed of a value and label, whereas in the latter case the interpretation will be of a value multiplied by a variable.

Disambiguation is also provided by displaying divisions and fractions in over/under form. Typing 61/3 produces:

$$\frac{61}{3}$$

Typing 6 1/3 produces:

$$6\frac{1}{3}$$

where the space following the 6 indicates that the "6" is a whole number and that the "1," followed by the "/," is the numerator of a fraction.

Several constraints are imposed upon entry to facilitate automatic analysis. The most troublesome, because of its potential effects on the problem-solving process, is that solutions must be expressed alpha-

numerically; no facility for entering diagrammatic representations is yet included (although examinees are free to draw such representations with paper and pencil as they work). Also as a consequence of requiring text, the solution will be expressed linearly and, almost assuredly, organized as a sequence. Third, certain syntax errors are detected and appropriate messages displayed. For example, when an equation is entered it is checked for unbalanced parentheses.

On the far right of the display is a five-function calculator (addition, subtraction, multiplication, division, square root), the availability of which is under experimenter control. If it is enabled, the calculator is used at the examinee's option. Entries are made by clicking on the operand and operator keys with the mouse or by using the numeric keypad. These actions produce a scrollable "tape" that comprises a recallable record of all entries and results.

Unlike some intelligent tutors that permit the examinee to click on an equation and have the result calculated automatically, in this system the examinee must key all values and reenter results in the workspace. This requirement was imposed to keep the complexity of the interface to a minimum and to be consistent with the way in which such problems are naturalistically solved: The examinee writes down the equation, keys the elements, and then copies the result. As we gain experience with how examinees use the calculator--and as naturalistic problem solving becomes more dependent on computer-based tools--we can add functions to take over these lower level actions.

Three primary icons are at the bottom of the screen. The Help icon displays a series of brief messages intended to explain generally the functions of the various interface components (see Figure 2). Each message appears next to its referent, and the messages are eliminated as soon as any key is pressed or the mouse is clicked. Clicking on Next Problem stores any entries made in the workspace and calculator, and presents the next item. Stop quits the test.

The interface now runs on a Macintosh 020/030/040-based computer with 4 megabytes of random access memory, a 13" monochrome display, and a standard keyboard. As the code has not yet been optimized, the Macintosh Fx is recommended. The Macintosh was chosen over the MS-DOS environment because collecting data on the Macintosh will permit solutions to be passed to GIDE for analysis without the added step of converting to another operating system format. Additionally, having the presentation and analysis components on the same machine brings us very close to integrating them in a performance assessment prototype that produces real-time analyses. Once that prototype is produced, we can consider implementing an operational version in the OSA environment on MS-DOS machines.

Smalltalk V (Digitalk,Inc.) and a related graphics tool, Widgets (Acumen Software), were used to program the interface. Smalltalk is an "object-oriented" programming language for developing easily extensible systems. Smalltalk code is highly portable and provides more graphical capability than such rapid prototyping tools as Hypercard. Smalltalk V, although somewhat less portable than other manufacturers' Smalltalk versions, offers a free run-

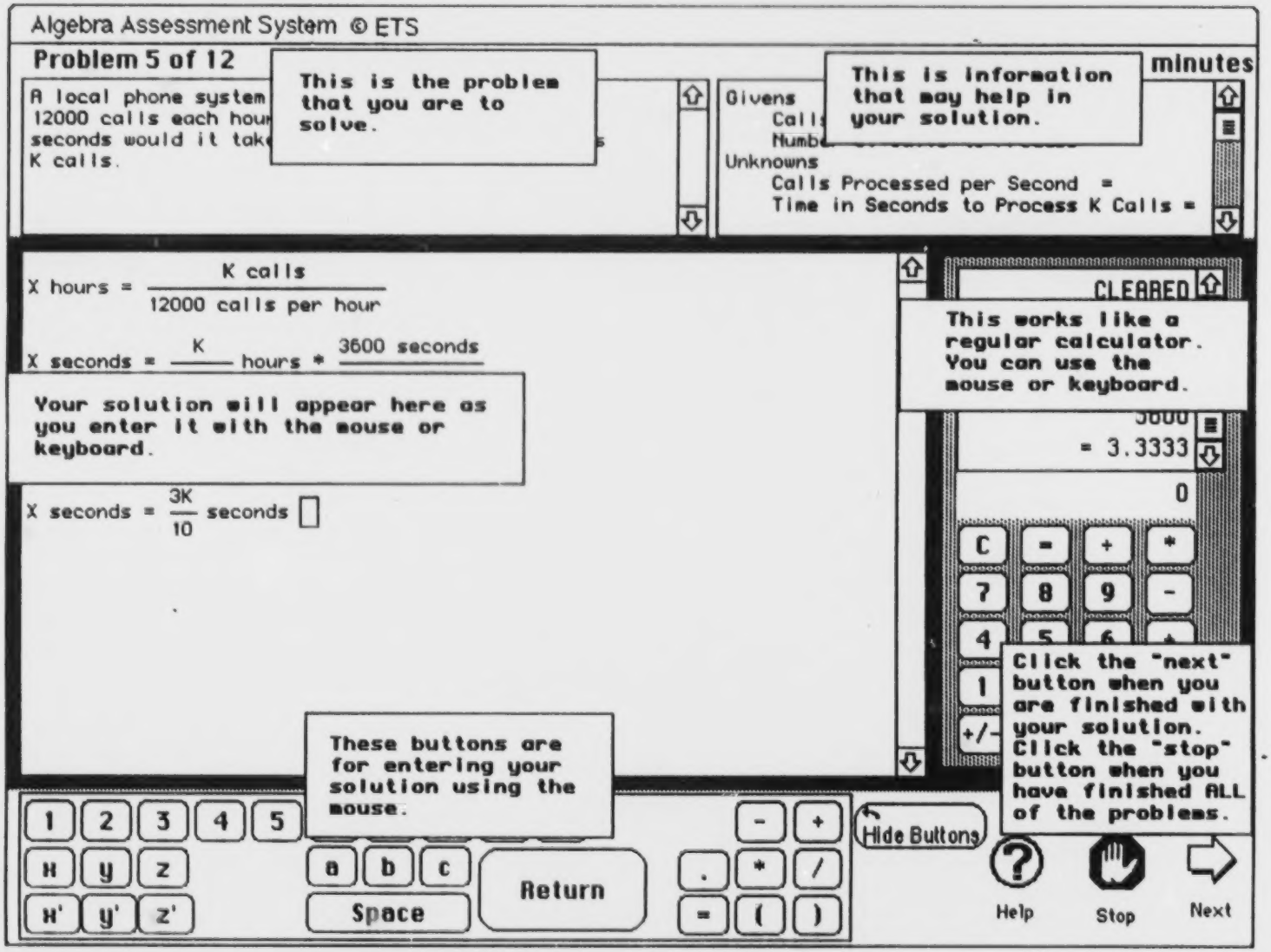

Figure 2. Interface with Help messages displayed.

time license, requires less run-time memory, and was substantially cheaper at initial purchase than competitors' implementations.

Future improvements. Pilot test results suggest that the interface functions effectively for research purposes. All the same, there are several issues that should be addressed in future work. First, pilot subjects generally noted differences between using the interface and solving problems in paper-and-pencil format. Subjects felt that paper and pencil was faster and allowed more flexibility, for example, the ability to draw diagrams or make tables. At the same time, when given the opportunity to use paper and pencil as a supplement, examinees chose not to do so. This choice may reflect the fact that when it is the computer input that is being evaluated, using paper-and-pencil may not be worth the added effort.

One means of increasing flexibility is to integrate a writing tablet with the system. Before pursuing this development, it would be worth determining more precisely how the absence of paper influences problem solving. Interestingly, the computer interface requires a more systematic solution approach than is the case for paper and pencil: The top-to-bottom, left-to-right progression inherent in text processing forces students to organize their solutions in a more orderly, step-by-step fashion. Additionally, typeface makes the solution easier to read. In contrast, paper-and-pencil solutions are often written so haphazardly and illegibly as to hamper the examinee's ability to organize, execute, and monitor them. The constraint imposed by the interface may, thus, improve performance for some individuals. Determining how, when, and for whom performance is affected should be systematically studied.

A second result of pilot testing was that examinees desired more sophisticated editing functions, particularly for cutting, copying, and pasting. The current interface was designed to require minimal computer experience and has few functions beyond simple data entry, editing, and calculation, which can generally be performed through either keyboard or mouse input. Future development should add more sophisticated options for the computer-literate examinee, without inhibiting novices. Such options might include enhanced editing functions as well as allowing transfer of information between the calculator and the workspace.

A third need is to provide more substantial explanation in how to use the interface. Improved explanation might be achieved through both a more sophisticated Help tool and a simple tutorial. The tutorial would teach system functions by providing problems with solutions for students to enter, tracking input, and intervening when functions were performed incorrectly.

Finally, the interface's capabilities could be extended beyond data collection to problem-solving support. One means for achieving this extension would be by using the format information window. When a problem was first presented, this window would be blank. If a wrong solution was posed, a general strategy for solving the problem might be displayed. If the revised solution was still incorrect, the problem might appear decomposed as a series of subproblems. A failure here might generate specific suggestions about how to carry out solution steps. An alternative to this "results-centered" approach would be to display information whenever the examinee requested it. Repeatedly clicking the mouse on the format window would display general

strategy, decomposition, and solution step information in sequence, with the examinee revealing only as much information as needed to solve the problem. The ability to call real-time analyses by GIDE would provide an additional support. These analyses would explain specifically what was wrong with the examinee's response and give a partial-credit score indicating how far the answer was from being correct.

Items and Knowledge Bases

As a result of our initial project (Sebrechts, Bennett, & Rock, 1991), GIDE had the knowledge to score and diagnostically comment upon solutions to 12 algebra word problems representing three classes (distance = rate x time, work, and interest) and four formats (open-ended, goal specification, equation setup, and faulty solution). (See Figure 3 for examples of the formats in their paper-and-pencil form and the upper right window of Figure 1 for an example in computer-based form.) Clearly, GIDE's utility, as proof that machine scoring of complex constructed responses is feasible and as the basis for an interactive diagnostic system, is enhanced as the number of problems it can analyze grows. As a consequence, we proposed to increase the number of problems GIDE could grade from 12 to 32 by adding isomorphic questions to the existing classes and a more difficult item and set of isomorphs from a new word problem class.

Four primary characteristics were sought in selecting the new prototype item: difficulty, decomposability, membership in a content class related to one of the existing classes, and prior use on the General Test. A more difficult problem was sought because the existing problems were found in our earlier studies to be considerably easier than anticipated (Sebrechts, Bennett, & Rock, 1991), and it is important to determine how adequately GIDE analyzes responses to difficult items. Decomposable problems--that is, problems that can be segmented into subtasks--were desired because substantial portions of GIDE's goal-plan analysis approach are predicated on this characteristic. Third, a content class conceptually close to one of the existing classes was important because it would make knowledge base development more efficient. Finally, prior use on the General Test provided easy access to difficulty estimates and maintained a direct substantive link with the testing program.

In our earlier work, we selected 20 algebra word problems from the General Test and administered them in open-ended form to a sample of undergraduates who produced written and verbal protocol data (Sebrechts, Bennett, & Rock, 1991). To select the new prototype problem, we used these data in conjunction with item statistics from GRE Program files. Of interest in reviewing these data is that very few of the most difficult items were decomposable. That is, the problems appeared to depend on a single equation, with the difficulty stemming from the insight required to detect the relationships among the givens and unknowns and algebraically formulate them in one key expression. This situation is in contrast to other problems in which the difficulty apparently stems from having to determine how the problem might be segmented into smaller, more easily solvable steps and from executing them. We know that goal-plan analysis can handle decomposable problems of easy-to-moderate difficulty quite successfully and suspect that it can also handle more difficult items. We are less sure about how effectively it can handle problems that are difficult but not decomposable. An incremental

Open-Ended

How many minutes will it take to fill a 2,000-cubic-centimeter tank if water flows in at the rate of 20 cubic centimeters per minute and is pumped out at the rate of 4 cubic centimeters per minute?

_____
_____
_____
_____

ANSWER:_____

Goal

One of two outlets of a small business is losing $500 per month while the other is making a profit of $1,750 per month. In how many months will the net profit of the small business be $35,000?

Givens
Profit from Outlet 1 = _____
Profit from Outlet 2 = _____
Target Net Profit = _____

Unknown
Net Monthly Profit = _____
= _____
Months to Reach Target Net Profit = _____
_____

ANSWER:_____

Equation Setup

A specialty chemical company has patented a chemical process that involves 2 reactions. Reaction 1 generates 24 grams of molecule B per minute and reaction 2 consumes 5 grams of molecule B per minute. If 4,560 grams of molecule B are desired as a product of this process, how many minutes must it continue?

Equations That Will Provide a Solution:

Net Amount of B Per Minute = Amount. Produced by Reaction 1 + Amount. Produced by Reaction 2
Time for Desired Amount of B = Desired Amount of B/Net Amount of B Per Minute

Your Solution:

_____
_____
_____
_____

ANSWER:_____

Faulty Solution

$3.50 in tolls is received each minute at an automated toll booth while the rate at a booth with an operator is $2.80 each minute. How many minutes elapse before the automated booth receives $14.00 more in tolls than does the person-operated booth?

Tolls per Minute = $3.50/min + $2.80/min
Tolls per Minute = $6.30/min
Time for $14 lead = $14/$6.30 per minute
Time for $14 lead = 2.22 minutes

Your Corrected Solution:

_____
_____
_____
_____

ANSWER:_____

Note. From Machine-scorable complex constructed-response quantitative items: Agreement between expert system and human raters' scores (RR-91-11) by M. M. Sebrechts, R. E. Bennett, and D. A. Rock. (Princeton, NJ: Educational Testing Service, 1991).

Figure 3. Isomorphs in four item formats.

approach to these issues suggests that difficulty-with-decomposability be addressed first. If responses to such problems can be successfully processed, the more challenging issue of difficulty-without-decomposability can be approached.

The limited overlap of decomposability and difficulty made it hard to find an item that met all four selection criteria (i.e., difficulty, decomposability, new content class, prior use on the General Test). Consequently, we decided to select two prototypes, both of which had been used on the General Test and were decomposable: a more difficult item from a previously examined content class, distance = rate x time (DRT), and a relatively easy item from a new content class, graduated rate (see Table 1). Difficulty statistics for the DRT problem were an equated delta of 16.0 and p+ of .22. The statistics for the graduated rate item were 9.9 and .74, respectively.

Table 2 is a blueprint of the current item pool. Each of the 40 problems has been cast in all four constructed-response formats. The complete pool is given as Appendix A.

To analyze students' solutions, GIDE draws on a knowledge base, in essence an understanding of how the problem is correctly and incorrectly solved. GIDE's knowledge base is specific to a problem and its isomorphs. For each such class, the knowledge base may contain several problem decompositions, where a decomposition is an organization of the problem into goals that must be solved along the path to generating a correct response. For each goal in turn, the knowledge base typically contains multiple correct and faulty implementations, or plans. GIDE analyzes a student's production goal by goal, comparing plans from its knowledge base with pieces of the student's solution. This goal-plan analysis strategy gives GIDE considerable leverage: Correct and incorrect implementations can be put together in different combinations to handle the variety of responses generated by examinees.

Our initial work with GIDE produced knowledge bases for the original DRT, work, and interest problem classes. These knowledge bases contained canonical solutions to the problems from each class, a catalogue of specific errors, and score penalties for the errors. Errors were grouped into four major categories:

> Mathematical errors involved a failure to execute a low-level operation (e.g., by inappropriately shifting a decimal, by incorrectly treating the remainder of a division as a decimal).
>
> Specific plan errors were inappropriate procedures for solving a goal linked to a particular problem class (e.g., confusing the rates for different trip segments).
>
> General plan errors suggested more universal failures to formulate procedures, with the same malformation having the potential to occur across problem contexts (e.g., dividing when multiplication is called for).
>
> Missing goals suggested the omission of a critical solution component.

Table 1

New Prototype Problems

1. A car is traveling at an average speed of 80 kilometers per hour. On the average, how many seconds does it take the car to travel K kilometers?

(A) $\frac{K}{80}$ (B) $\frac{K}{45}$ (C) $\frac{4K}{3}$

(D) 45K (E) 288,000K

2. A lawyer charges \$100 for the first hour of service and \$75 for each additional hour. A bill of \$625 represents how many hours of the lawyer's services?

(A) 7

(B) $7\frac{1}{2}$

(C) 8

(D) $8\frac{1}{2}$

(E) 9

Table 2

Blueprint of the Algebra Word Problem Pool

Item Category	Content Class				
	DRT	Work	Percent	Graduated Rate	Total
Existing GRE Prototype	1	1	1	0	3
New GRE Prototype	1	0	0	1	2
Existing Isomorph	3	3	3	0	9
New Isomorph	11	4	4	7	26
Total	16	8	8	8	40

Note. Each problem appears in four formats.

Points were deducted depending on the number and types of errors found in the solution: Mathematical errors cost one point; general and specific plan errors one point if computational, two if the plan structure was affected; and missing goal bugs three points. Score scales for an item were based on the number of goals that comprised the most commonly used canonical solution. (See Appendix B for the original rubric.)

Similar knowledge bases were needed for the new DRT (DRT-2) and graduated rate problems. As for the original knowledge bases, these structures were built using cognitive task analysis. Because both the DRT-2 problem and the graduated rate prototypes came from the 20-item pool used in our initial algebra study (Sebrechts, Bennett, & Rock, 1991), data for the cognitive analysis were already in hand. Also, these data had been partially analyzed in connection with a project to build a cognitive framework for algebra problem solving. In combination, these factors made it feasible to develop knowledge bases for the two problem classes (instead of the single class originally intended).

Two data sets were analyzed. The first set was provided by content experts (i.e., ETS mathematics test development staff) and consisted of their descriptions of how the prototype problems could be decomposed and what wrong solution paths examinees might take. The second data set was comprised of protocols provided by 51 undergraduates attending The Catholic University of America. The information gathered from this analysis was then coded for GIDE to use in processing student responses.

Table 3 gives the canonical solutions and Table 4 the errors abstracted from the cognitive analysis for use in the knowledge bases. As Table 3 indicates, the graduated rate problem is decomposed into three goals and the DRT-2 problem into two goals. These decompositions and errors supplement those for the original three problem classes (see Appendix C). Although a detailed discussion of the new solutions and errors is beyond the scope of this report, several observations are appropriate.

First, as for the previous problems, the canonical solutions are equation based, in keeping with the representations that GIDE was designed to process. The practical effect of this constraint has so far been minimal: In the knowledge-base development sample, the vast majority of student paper-and-pencil solutions to the new problems took an equation format that could be analyzed by GIDE. So, even though the graduated rate problem can be solved by simulation, only a small number of students chose that option. GIDE can accurately process these responses if they are either totally wrong or completely correct by examining the end result. It can also analyze simulations that are partially correct if they produce correct intermediate products along the canonical solution path(s). It does this processing through "implicit matching," which allows it to infer, absent evidence to the contrary, the integrity of initial and intermediate solution steps from the adequacy of subsequent ones. Because GIDE does not logically follow the substance of the simulations (as it does equation-based solutions), it cannot accurately analyze simulations that are partially wrong and do not use a decomposition similar to one of the canonical equation-based solutions. The DRT-2 problems, in contrast, are not well suited to simulation because they contain a constant. More likely is a ratio approach. Because ratio solutions take an equation form, GIDE can usually analyze these responses accurately.

Table 3

Canonical Solutions for Two New Problem Classes

Graduated Rate

A lawyer charges $100 for the first hour of service and $75 for each additional hour. A bill of $625 represents how many hours of the lawyer's service?

Canonical Solution:

1. Lawyer's Charge After First Hour = $625 - $100
   Lawyer's Charge After First Hour = $525
2. Number of Additional Hours = $525 / $75
   Number of Additional Hours = 7 hours
3. Total Number of Hours = 7 hours + 1 hour
   Total Number of Hours = 8 hours

Distance = Rate x Time II (DRT-2)

A car is traveling at an average speed of 80 kilometers per hour. On the average, how many seconds does it take the car to travel K kilometers?

Canonical Solutions:

A 1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
   Kilometers Traveled per Second = 1/45 Kilometers per Second
  2. Seconds to Travel K Kilometers = K Kilometers / 1/45 Kilometer per Second
   Seconds to Travel K Kilometers = 45K Seconds

B 1. Hours to Travel K Kilometers = K Kilometers / 80 Kilometers per Hour
   Hours to Travel K Kilometers = K/80 Hours
  2. Seconds to Travel K Kilometers = K/80 Hours * 3,600 Seconds per Hour
   Seconds to Travel K Kilometers = 45K Seconds

C 1. Seconds per Kilometer = 1/80 Hour per Kilometer * 3,600 Seconds per Hour
   Seconds per Kilometer = 45 Seconds per Kilometer
  2. Seconds to Travel K Kilometers = 45 Seconds per Kilometer * K Kilometers
   Seconds to Travel K Kilometers = 45K Seconds

Note. Letters identify different canonical solutions. Line numbers identify goals that comprise a solution.

Table 4

Errors Represented in Knowledge Bases for Two New Problem Classes

Mathematical Errors

Division Operator with Multiplication Operation (109): A correct division plan is used, but the computation performed is a multiplication. This error is most likely to occur with fractional denominators.

DRT-2 1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = K Kilometer / 1/45 Kilometers per Second
Seconds to Travel K Kilometers = K/45 Seconds

Specific Plan Errors for Graduated Rate

Use first rate as only rate (802): The rate for the first unit of time is assumed to be the rate for the entire time.

1. Number of Hours = $625/ $100 per Hour = 6.25 Hours

Use second rate as only rate (803): The rate for the period after the first unit of time is assumed to be the rate for the entire time.

1. Number of Hours = $625/ $75 per Hour = 8.33 Hours

Rate Difference as Average Cost (804): The difference between first unit and subsequent rates is taken as the average rate for the entire time period.

1. Rate = $100 per Hour - $75 per Hour
Rate = $25 per Hour
2. Time = $625 / $25 per Hour
Time = 25 Hours

Rate Average as Hourly Cost (805): The average of the first unit and subsequent rates is taken as the average rate for the entire time period.

1. Rate = ($100 per Hour + $75 per Hour) / 2
Rate = $87.50 per Hour
2. Time = $625 / $87.50 per Hour
Time = 7.14 Hours

Note. Line numbers indicate goals keyed to Table 3 canonical solutions. All DRT-2 errors are keyed to canonical solution A in Table 3.

Table 4

Errors Represented in Knowledge Bases for Two New Problem Classes

Specific Plan Errors for Distance = Rate x Time (2)

Convert to Minutes Instead of Seconds (307): In making a time conversion the change is made to minutes instead of to seconds.

1. Kilometers Traveled per Second = 80 Kilometers per Hour / 60 Seconds per Hour
   Kilometers Traveled per Second = 1.33 Kilometers per Second
2. Seconds to Travel K Kilometers = K Kilometers / 1.33 Kilometers per Second
   Seconds to Travel K Kilometers = 0.75K Seconds

Invert Time Conversion (308): The rate is multiplied by the time conversion instead of divided by the conversion.

1. Kilometers Traveled per Second = 80 Kilometers per Hour * 3,600 Seconds per Hour
   Kilometers Traveled per Second = 28,880 Kilometers per Second
2. Seconds to Travel K Kilometers = K Kilometers / 28,880 Kilometers per Second
   Seconds to Travel K Kilometers = 28,880K Seconds

Time = Distance x Rate (415): The incorrect formula T=DR is used in place of T=D/R.

1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
   Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = K Kilometers * 1/45 Kilometers per Second
   Seconds to Travel K Kilometers = K/45 Seconds

Time = Rate / Distance (416): The incorrect formula T=R/D is used in place of T=D/R.

1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
   Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = 1/45 Kilometers per Second / K Kilometers
   Seconds to Travel K Kilometers = 1/45K Seconds

Note. Line numbers indicate goals keyed to Table 3 canonical solutions. All DRT-2 errors are keyed to canonical solution A in Table 3.

Table 4

Errors Represented in Knowledge Bases for Two New Problem Classes

Specific Plan Errors for Distance = Rate x Time (2)

Treat Constant as 1 (731): The problem is solved as though the constant were equal to 1.

DRT-2 1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = 1 Kilometer / 1/45 Kilometers per Second
Seconds to Travel K Kilometers = 45 Seconds

Use Time Conversion as Time (732): The value of the conversion is used as the actual time.

DRT-2 1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = 1/45 Kilometers

Missing Goal Errors for Distance = Rate x Time (2)

Missing First Goal (951): Rate of travel in converted time units is missing. The goal takes the canonical form Rate of Travel in Converted Units = Rate of Travel in Base Unit / Unit Conversion.

2. Seconds to Travel K Kilometers = K Kilometers / 80 Kilometers per Hour
Seconds to Travel K Kilometers = K/80 Seconds

Missing Second Goal (952): Time to travel is missing. The goal takes the canonical form Time to Travel Specified Distance = Distance to Travel / Rate of Travel.

1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
Kilometers Traveled per Second = 1/45 Kilometers per Second

Note. Line numbers indicate goals keyed to Table 3 canonical solutions. All DRT-2 errors are keyed to canonical solution A in Table 3.

Table 4

Errors Represented in Knowledge Bases for Two New Problem Classes

Missing Goal Errors for Graduated Rate

Missing First Goal (941): Charge After First Unit of Time is missing. The goal takes the canonical form, Charge After First Unit of Time = Total Charge - Charge for First Unit of Time.

2. Number of Additional Hours = $625 / $75
   Number of Additional Hours = 8.33 Hours
3. Total Number of Hours = 8.33 Hours + 1 Hour
   Total Number of Hours = 9.33 Hours

Missing Second Goal (942): Time at Subsequent Rate is missing. The goal takes the canonical form, Time at Subsequent Rate = Cost at Subsequent Rate / Subsequent Rate. This bug occurs only when #943 is also present.

1. Lawyer's Charge After First Hour = $625 - $100
   Lawyer's Charge After First Hour = $525

Missing Third Goal (943): Total Time is missing. The goal takes the canonical form, Total Time = Time at Base Rate + Time at Subsequent Rate. The time after the initial rate is used, ignoring the addition of the base unit (usually the charge for 1 hour, 1 mile, etc.)

1. Lawyer's Charge After First Hour = $625 - $100
   Lawyer's Charge After First Hour = $525
2. Number of Hours = $525 / $75
   Number of Hours = 7 Hours

Note. Line numbers indicate goals keyed to Table 3 canonical solutions.

A second observation is that we were able to build on the knowledge bases for the existing problem classes in constructing knowledge bases for the new problems. For example, the extant math and general-plan bugs were used as is because they were designed to capture low-level errors in addition, subtraction, and multiplication that can occur in any problem. In those cases in which the cognitive analysis suggested a possible, but highly unlikely, bug, existing general plan errors were used instead of adding specific ones. Thus, for the graduated rate problems, it is possible to add the initial hour's cost to the total cost instead of finding the difference; however, having never observed such an error in student data, it is left to be handled by the general-plan Add for Subtract bug (205). This strategy makes for a more concise matching mechanism overall and for greater generality in expanding to new problems.

Whereas portions of the old knowledge bases could be used, the new problems did, of course, require novel plans. For example, the DRT-2 problem can be solved by calculating the time the car needs to travel using the equality Time = Distance / Rate. Some students instead use the formula Time = Distance * Rate. A specific-plan bug, Time = Distance x Rate (415), has been added to distinguish this time-related error from a more general confusion in the use of operators, which is trapped by the general-plan bug, Times for Divide (203). Other students calculate the time using the correct division formulation but erroneously execute it by multiplying the values. A new math bug, Division Operator with Multiplication Operation (109), was introduced to trap this error (see Table 4 for an example).

Third, the new knowledge bases suggest that the scoring rules used by GIDE may need to be modified. The current approach bases the score range on the number of goals comprising an item's most commonly used canonical solution, with three points awarded for achieving each goal. This scheme was intended to reflect problem difficulty, with the number of goals acting as an approximate indicator. The DRT-2 problem makes clear that this indicator is imperfect: It has two goals but appears considerably more difficult than the original five-goal DRT problem. (The GRE General Test difficulty statistics for the two original prototype problems support this assertion.) It is possible that partial-credit modeling of these scores can adjust for these differences, but a more straightforward solution might be to modify the scoring framework.

A fourth observation is that although the new knowledge bases provide reasonable coverage of the solution paths and errors found in our cognitive analysis, GIDE will occasionally experience difficulty processing new solutions. One instance relates to the terminology used to designate units. Although GIDE can recognize several optional spellings and abbreviations (e.g., secs, seconds, sec, sec., secs.), it cannot anticipate all the alternatives (e.g., scnd). (Any term can be easily added, however.) Further, in some instances the terminology used may overlap with other solution components, adding complexity to the analytic task. This difficulty is especially pronounced in the DRT-2 K Kilometers problem. If the student chooses to use K to also indicate kilometer, GIDE may confuse the abbreviation with the constant. This difficulty could be eliminated easily enough by using another letter to represent the constant. For the moment, we have chosen to retain the problem as is because this is its original GRE General Test form

and because we are interested in investigating the contribution to problem difficulty of working with K rather than, say, T kilometers.

Another instance in which GIDE may experience difficulty concerns the identification of errors. Difficulty in correctly isolating errors is a function of the potential solution space: As the number of solution paths increases, overlap is more likely and it becomes harder to identify which path a student intended to traverse. The DRT-2 problem provides an example. Here, the meaning of a given error depends on the solution path intended by the student and where along that path the error occurs. Identifying the particular path and goal on which a student is working can be especially difficult for incomplete solutions (although ambiguities can arise in full solutions because students do not always use labels that would clarify the solution structure). For example, one approach to solving the DRT-2 problem is:

1. Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
   Kilometers Traveled per Second = 1/45 Kilometers per Second
2. Seconds to Travel K Kilometers = K Kilometers / 1/45 Kilometer per Second
   Seconds to Travel K Kilometers = 45K Seconds

The partial solution, X = 80 * 3,600, would indicate that the examinee had attempted the first goal and used the wrong operator in making the conversion from hours to seconds. This same partial solution, however, would have a different (though still incorrect) meaning in the following alternative approach:

1. Seconds per Kilometer = 1/80 Hour per Kilometer * 3,600 Seconds per Hour
   Seconds per Kilometer = 45 Seconds per Kilometer
2. Seconds to Travel K Kilometers = 45 Seconds per Kilometer * K Kilometers
   Seconds to Travel K Kilometers = 45K Seconds

Here, the first goal is to compute not the rate, but its inverse, the seconds per kilometer. Consequently, the partial solution, X = 80 * 3,600, can be interpreted as a correct multiplicative structure for solving the goal but with an inversion error (80 hours instead of 1/80 hour).

In addition to producing the two new knowledge bases, the three existing ones were modified to support the new isomorphs for the DRT, work, and interest problem classes. The time required to make these modifications was minimal, about two days per isomorph. This level of effort could be dramatically reduced through automation. Such a technique would take the givens of a new isomorph and modify the existing knowledge base by substituting the appropriate values, units, variables, and constants. Building this automatic capability would probably require a person-week of effort; once built, the knowledge-base modifications for an isomorph could be produced in under an hour, plus some additional time for adjusting the content of the bug messages to fit the isomorph.

Inference Mechanism Improvements

Two minor faults in GIDE's inference mechanism were corrected. One fault pertained to unit labels. In solving a given problem, some students always use labels, some leave them off entirely, and some use them inconsistently. GIDE uses units in its analysis whenever they are present in the solution. When not explicitly stated, GIDE assumes the student is using the units specified by the problem stem (unless the solution presents evidence to the contrary). This analytic strategy worked well except when students solved the problem correctly in different units (e.g., minutes instead of hours) and failed to use labels consistently. In these instances, GIDE correctly processed the stated labels when attached to specific values, but switched to the problem-specified units when labels were absent, causing some pieces of the solution to be erroneously interpreted. This fault was corrected by introducing a mechanism that checks whether a plan would be correct if the units were transformed to other acceptable units. Thus, if a solution component is not correct when interpreted in terms of hours, GIDE will process the units as minutes and reevaluate the plan. This mechanism provides a reasonable resolution in that it will accurately process most correct formulations. Wrong formulations will generally be treated as missing goals. The specific errors present in a plan in which the student uses different units but fails to label them will not be identified by this mechanism.

A second inference fault concerned GIDE's interpretation of wrong results in the context of correctly structured plans. This fault appeared to account for GIDE's tendency, detected in our previous research, to award slightly lower scores on average than did human raters (Sebrechts, Bennett, & Rock, 1991). Correct problem solutions sometimes include intermediate results that are close to one another. For example, the "600 mile" DRT problem has one distance segment traveled in 6.3 hours and another in 6.33 hours (see Appendix A, DRT-17). For scoring and error diagnosis to be implemented properly, GIDE needs to associate each of these values with the appropriate goal. One way it does this association is by imposing value constraints, that is, comparing incorrect values to a range of "reasonableness" built around the expected result. In our previous research, these ranges tended to be more restrictive than those applied by the raters, sometimes causing GIDE to infer that a goal was missing (a 3-point deduction) or structurally incorrect (2-point deduction) when it was more properly considered a computational error (1-point deduction). This inference fault was corrected by instituting a more general mechanism for determining the range of appropriate values and by slightly relaxing the constraints on what values are acceptable for a given goal.

Discussion

This project built an interface for interactively presenting algebra word problems and collecting constructed responses in machine-readable form. Although the interface was built to function with GIDE, it was designed to write a generic output file that could be used with other automatic analysis programs. Second, the project increased GIDE's content coverage by adding items and the knowledge bases required for response analysis. Finally, GIDE's inference mechanism was enhanced. Together, these improvements constitute a research platform--the Algebra Assessment System--that permits stronger

inferences to computer-based testing environments and forms the basis for an eventual interactive performance assessment prototype.

What computer-based testing research does this platform make feasible? One important issue is how the constraints imposed in computer-based testing affect problem solving. As noted, a key concern is to design interface constraints that facilitate computer delivery and analysis but that do not materially affect the nature of problem solving. The Algebra Assessment System might be used to investigate the effects of several such constraints including availability of computational aid, mode of data entry, and solution form. The availability of the on-screen calculator, for example, can be manipulated to determine if its use produces a purer measure of mathematical problem solving by reducing the likelihood of low-level computational mistakes. Different data entry modes (keyboard, mouse, mouse and keyboard) can be tried to see if one or another mode introduces extraneous difficulty. Finally, the constraints imposed on solution form can be assessed by dividing examinees among interface, interface in conjunction with pencil-and-paper, and paper-and-pencil only groups.

In addition to interface constraints, some aspects of problem format should work to influence analytical accuracy. We have not been able to document such an influence when the formats encourage solutions of equivalent complexity. So, for example, GIDE is no more accurate in scoring a response to an equation setup or faulty solution item than to one presented in open-ended format, possibly because in each format examinees generate the full solution. Formats that require fewer solution elements, however, should be easier to score than those calling for many components. Braun, Bennett, Frye, and Soloway (1990) found that a computer science version of the faulty solution item that required only insertion and deletion could be scored more accurately than could complete programs produced ab initio. We might get a rough indication of how dramatically accuracy can be improved in our algebra item pool simply by processing different portions of existing solutions. For instance, one scoring run might process the entire solution, including the plan structures and the intermediate and end results; a second run only the intermediate and end results; and a final run only the end result. A second approach might be to collect new data varying the extent of the required response. Some examinees might be required to show all work, some to produce only the intermediate results and the final answer, some to fill missing slots in an incomplete solution (or make insertions and deletions directly into a wrong solution), and some to produce only the expression or value that constitutes the final result.

Third, with some minor changes to the interface's format window, we can study the effect of providing progressively more refined information about the problem decomposition. As described, this facility in its simplest form would allow the examinee to request help in solving the problem. Of interest not only as a problem-solving support, this facility also can be used to study the value of "dynamic assessment" (Feuerstein, 1979) for graduate admissions. Dynamic assessment attempts to directly measure scholastic ability by presenting a mini-learning situation. In this situation, a problem is presented that is known to be just beyond the examinee's current reach. After this pretest, hints are given and the examinee is retested. Finally, a transfer task may be posed to determine if knowledge gained from the original situation can be applied in a new context. Several ability measures can be

derived from such a session, including the extent of assistance, degree of improvement, attainment after help, and achievement on the transfer task. In the Algebra Assessment System, the extent of assistance might be summarized by the number of requests made for each item or by how far down in the decomposition hierarchy the requests reach (i.e., the extent to which general versus specific feedback is solicited). The raw data for evaluating the examinee's improvement could be produced by saving the solution upon each request and then scoring each member of the progression offline. Near transfer could be evaluated using problem isomorphs. Indicators such as these might enhance prediction of graduate performance beyond that provided by the General Test. Campione and Brown (1987) and Embretson (1987a, 1992), among others, have presented data to support the validity of similar indicators as predictors of cognitive performance. Embretson (1987b, 1992) has also suggested approaches for handling the unique psychometric problems these indicators may pose.

What products might evolve from this platform? One product is an instructional assessment system for helping undergraduates, in particular educationally disadvantaged ones, identify and ameliorate weaknesses in the quantitative reasoning skills needed for success in graduate school. Such a system would eventually need to have the same content breadth as the General Test, including real and abstract problems in arithmetic, algebra, and geometry. The Algebra Assessment System can handle a subset of algebra word problems. Because arithmetic word problems require essentially similar analytical capabilities, the system can handle many of these items too (once the proper knowledge bases have been added). These two problem types should offer enough coverage to test the utility of the practice-system notion in an undergraduate context. If the concept appears useful, the facility to analyze responses to pure problems, to capture and analyze diagrammatic solution components, to process simulations, and to handle geometry problems might be progressively added.

Two extensions of this work with potential implications for the General Test should be mentioned. One concerns our decision to take a knowledge-based approach to response analysis. This decision stemmed from the nature of word problems, the responses to which cannot be evaluated without some understanding of the relations between a problem's givens and goals; responses to word problems can be algebraically and computationally correct without solving the problem, so that an analysis system must in some sense understand the problem if it is to process solutions accurately. In GIDE's case, this understanding takes the form of one or more decompositions and a series of correct and incorrect plans. This knowledge allows it to recognize specific errors and, from these errors, generate a partial-credit score. If, as in the General Test, detailed item-level error diagnosis is not desired, scoring can be approached more directly, at least for pure algebra problems that call for procedural solutions. For instance, consider a constructed-response version of the following General Test problem:

If $3x + 5 = 2x + 10$, then $x =$

(Educational Testing Service, 1988, p. 105).

Open-ended responses to problems like this can be scored by an analyzer, the core of which would be a symbol manipulator that solves the problem and

compares its solution process and end result to the examinee's production. Symbol manipulators that solve far more complex procedural problems are readily available, found in such off-the-shelf software as Theorist (Prescience Corporation), a mathematical problem-solving tool. The major efficiency gained over the knowledge-based approach is that, once developed, such a routine could score responses to any procedural problem whose required transformations were within the set that the manipulator could apply. If it is desirable in future versions of the computer-delivered General Test to capture and score the solution process as well as the end result, this approach might be worth investigating.

A second extension concerns the focus of problem solving. GIDE's strength is in identifying how the plans an examinee uses to solve a word problem differ in their structure or execution from a canonical solution. The problem-solving process, of course, involves additional components, not the least of which is problem representation. Considerable evidence exists to suggest that experts represent problems by category and that these categories direct problem solving (Chi, Feltovich, & Glaser, 1981; Glaser, 1991). Categories are thought to direct problem solving by eliciting a knowledge structure or schema that, at least for experts, includes potential solution methods. The initial categorization restricts search to a small range of potential solution paths. A classic method of assessing representational skill is the categorization task: Present a set of word problem stems that vary in both underlying solution structure and their surface features, or "cover stories," and ask the examinee to classify the problems according to mathematical structure. Such a task can be computer delivered, objectively scored without resort to costly knowledge-based methods, and efficiently completed (the problems need only be classified, not solved). Consequently, it might be worth incorporating this task into the Algebra Assessment System, thereby offering an opportunity to study its utility. For example, some examinees may be more adept at representing problems relative to planning and executing their solutions. Others may be able to construct an adequate representation only as a result of attempting to execute several inappropriate plans. Such cognitive differences have obvious instructional implications but also might afford predictive information beyond that contained in GRE quantitative scores.

Along with ETS, the GRE Program has shown considerable interest in developing computer-based testing services. For these services to be successful, they must provide distinct improvements over existing tests in the tasks posed and the information returned. Interactive performance assessment represents one direction that may offer such improvements. As a crude exemplar, the Algebra Assessment System gives some initial ideas as to what this direction might hold and how those ideas might be achieved through further research and development.

References

Bennett, R. E. (in press). Toward intelligent assessment: An integration of constructed response testing, artificial intelligence, and model-based measurement. In N. Frederiksen, R. J. Mislevy, and I. Bejar (Eds), Test theory for a new generation of tests. Hillsdale, NJ: Erlbaum.

Bennett, R. E. (in press). Environments for presenting and automatically scoring complex constructed responses (RM-92-5). Princeton, NJ: Educational Testing Service.

Bennett, R. E., Sack, W., & Soloway, E. (1991). A practice and feedback system for Advanced Placement Computer Science (RM-92-9). Princeton, NJ: Educational Testing Service.

Bennett, R. E., Sebrechts, M. M., & Rock, D. A. (1991). The convergent validity of machine-scorable, constructed-response GRE General Test quantitative items (RR-91-12). Princeton, NJ; Educational Testing Service. (Also Applied Psychological Measurement, 1991, 15, 227-239.)

Bennett, R. E., Sebrechts, M. M., & Yamamoto, K. (1991). Fitting new measurement models to GRE General Test constructed-response item data (RR-91-60). Princeton, NJ: Educational Testing Service.

Braun, H. I., Bennett, R. E., Frye, D., & Soloway, E. (1990). Scoring constructed responses using expert systems. Journal of Educational Measurement, 27, 93-108.

Brown, J. S. (1985). Process versus product: A perspective on tools for communal and informal electronic learning. Journal of Educational Computing Research, 1, 179-201.

Campione, J. C., & Brown, A. L. (1987). Linking dynamic assessment with school achievement. In C. S. Lidz (Ed.), Dynamic assessment: An interactional approach to evaluating learning potential. New York: The Guilford Press.

Chi, M. T., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. Cognitive Science, 5, 121-152.

Educational Testing Service. (1988). Practicing to take the GRE General Test--No. 6. Princeton, NJ: Author.

Embretson, S. E. (1987a). Improving the measurment of spatial aptitude by dynamic testing. Intelligence, 11, 333-358.

Embretson, S. E. (1987b). Toward development of a psychometric approach. In C. S. Lidz (Ed.), Dynamic assessment: An interactional approach to evaluating learning potential. New York: The Guilford Press.

Embretson, S. E. (1992). Measuring and validating cognitive modifiability as an ability: A study in the spatial domain. Journal of Educational Measurement, 29, 25-50.

Feuerstein, R. (1979). The dynamic assessment of retarded performers: The Learning Potential Assessment Device, theory instruments, and techniques. Baltimore: University Park Press.

Frederiksen, N., & Ward, W. C. (1978). Measures for the study of creativity in scientific problem-solving. Applied Psychological Measurement, 2(1), 1-24.

Glaser, R. (1991). Expertise and assessment. In M. C. Wittrock & E. L. Baker (Eds.), Testing and cognition. Englewood Cliffs, NJ: Prentice-Hall.

Kaplan, R. (1992). Using a trainable pattern-directed computer program to score natural language item responses (RR-91-31). Princeton, NJ: Educational Testing Service.

McArthur, D., Stasz, C., & Hotta, J. Y. (1986-87). Learning problem-solving skills in algebra. Journal of Educational Technology Systems, 18, 303-324.

Mislevy, R. J. (in press). Foundations of a new test theory. In N. Frederiksen, R. J. Mislevy, and I. Bejar (Eds.), Test theory for a new generation of tests. Hillsdale, NJ: Erlbaum.

Resnick, L. B., & Resnick, D. P. (1990). Tests as standards of achievement in schools. In Proceedings of the 1989 ETS Invitational Conference: The uses of standardized tests in American education (pp. 63-80). Princeton, NJ: Educational Testing Service.

Sebrechts, M. M., Bennett, R. E., & Rock, D. A. (1991). Machine-scorable complex constructed-response items: Agreement between expert system and human raters' scores (RR-91-11). Princeton, NJ: Educational Testing Service. (Also Journal of Applied Psychology, 1991, 76, 856-862.)

Sebrechts, M. M., LaClaire, L., Schooler, L. J., & Soloway, E. (1986). Toward generalized intention-based diagnosis: GIDE. In R. C. Ryan (Ed.), Proceedings of the 7th National Educational Computing Conference (pp. 237-242). Eugene, OR: International Council on Computers in Education.

Shepard, L. (1991). Interview on assessment issues with Lorrie Shepard. Educational Researcher, 20(2), 21-23, 27.

Singley, J. K., Anderson, J. R., & Gevins, J. S. (1990). Promoting abstract strategies in algebra word problem solving (IBM Research Report 15861). Yorktown Heights, NY: IBM T. J. Watson Research Center.

Wenger, E. (1987). Artificial intelligence and tutoring systems. Los Altos, CA: Morgan Kaufmann.

Appendix A

Item Pool

The item pool is organized into five problem classes named for the prototypes on which the classes are based: work, interest, distance = rate x time, graduated rate, and distance = rate x time (2). The items within each class are isomorphic, although their cover stories may no longer reflect the surface features normally associated with a given content class. Each item is followed by one or more canonical solutions. Items can be presented in any one of four formats:

Open-ended provides only the stem; the examinee must offer a correct solution.

Goal-specification lists the labels for the givens and unknowns but does not provide the actual values.

Equation-setup provides a set of general equations that would solve the problem. If the problem has more than one canonical solution, these equations are based on the first solution.

Faulty-solution consists of a variant on the correct solution that incorporates an error, although the form of the solution is otherwise generally correct. If the problem has more than one canonical solution, the faulty answer is a variant of the first solution.

Work-1

How many minutes will it take to fill a 2,000 cubic centimeter tank if water flows in at the rate of 20 cubic centimeters per minute and is simultaneously pumped out at the rate of 4 cubic centimeters per minute?

Canonical Solution

Net Rate of Filling = 20 cc per minute - 4 cc per minute
Net Rate of Filling = 16 cc per minute
Time to Complete Filling = 2000 cc / 16 cc per minute
Time to Complete Filling = 125 minutes

Goal-Specification Format Information

Givens
Amount to Fill =
Rate Flowing In =
Rate Flowing Out =

Unknowns
Net Rate of Filling =
Time to Complete Filling =

Equation-Setup Format Information

Net Rate of Filling = Rate Flowing In - Rate Flowing Out
Time to Complete Filling = Amount to Fill / Net Rate of Filling

Faulty-Solution Format Information (sum instead of subtract rates: net rate = inflow rate - outflow rate => net rate = inflow rate + outflow rate)

Net Rate of Filling = 20 cc per minute + 4 cc per minute
Net Rate of Filling = 24 cc per minute
Time to Complete Filling = 2000 cc / 24 cc per minute
Time to Complete Filling = 83 1/3 minutes

Work-2

One of the two outlets of a small business is losing $500 per month while the other is making a profit of $1,750 per month. In how many months will the net profit of the small business be $35,000? (GS)

Canonical Solution

Net Monthly Profit = $1,750 per month - $500 per month
Net Monthly Profit = $1,250 per month
Months to Reach Target Net Profit = $35,000 / $1,250 per month
Months to Reach Target Net Profit = 28 months

Goal-Specification Format Information

Givens
Loss From Outlet 1 =
Profit From Outlet 2 =
Target Net Profit =

Unknowns
Net Monthly Profit =
Months to Reach Target Net Profit =

Equation-Setup Format Information

Net Monthly Profit = Profit From Outlet 2 - Loss From Outlet 1
Months to Reach Target Net Profit = Target Net Profit / Net Monthly Profit

Faulty-Solution Format Information (decimal shift error $35,000 => $3,500)

Net Monthly Profit = $1,750 per month - $500 per month
Net Monthly Profit = $1,250 per month
Months to Reach Target Net Profit = $3500 / $1,250 per month
Months to Reach Target Net Profit = 2.8 months

Work-3

A specialty chemical company has patented a chemical process that involves 2 reactions. Reaction 1 generates 24 grams of molecule B per minute and reaction 2 consumes 5 grams of molecule B per minute. If 4,560 grams of molecule B are desired as a product of this process, how many minutes must it continue?

Canonical Solution

Net Amount of B per Minute = 24 grams per minute - 5 grams per minute
Net Amount of B per Minute = 19 grams per minute
Time for Desired Amount of B = 4,560 grams / 19 grams per minute
Time for Desired Amount of B = 240 minutes

Goal-Specification Format Information

Givens
Amount Produced by Reaction 1 =
Amount Consumed by Reaction 2 =
Desired Amount of B =

Unknowns
Net Amount of B per Minute =
Time for Desired Amount of B =

Equation-Setup Format Information

Net Amount of B per Minute =
Amount Produced by Reaction 1 - Amount Consumed by Reaction 2
Time for Desired Amount of B = Desired Amount of B / Net Amount of B per Minute

Faulty-Solution Format Information (addition of separate rates as time: time = amount / avg. rate => time = rate1 + rate2)

Time for Desired Amount of B =
(4,560 grams / 24 grams per min.) + (4,560 grams / 5 grams per min.)
Time for Desired Amount of B = 190 minutes + 912 minutes
Time for Desired Amount of B = 1102 minutes

Work-4

$3.50 in tolls is received each minute at an automated toll booth while the rate at a booth with an operator is $2.80 each minute. How many minutes elapse before the automated booth receives $14.00 more in tolls than does the person-operated booth?

Canonical Solution

Difference in Toll Booth Rates = $3.50 per minute - $2.80 per minute
Difference in Toll Booth Rates = $0.70 per minute
Time for Desired Lead = $14 / $0.70 per minute
Time for Desired Lead = 20 minutes

Goal-Specification Format Information

Givens
Automated Booth Tolls per Minute =
Operator Booth Tolls per Minute =
Desired Amount of Monetary Lead =

Unknowns
Difference in Toll Booth Rates =
Time for Desired Lead =

Equation-Setup Format Information

Difference in Toll Booth Rates =
Automated Booth Tolls per Minute - Operator Booth Tolls per Minute
Time for Desired Lead =
Desired Amount of Monetary Lead / Difference in Toll Booth Rate

Faulty-Solution Format Information (multiplication for division: time = dollar lead / rate => time = dollar lead * rate)

Difference in Toll Booth Rates = $3.50 per minute - $2.80 per minute
Difference in Toll Booth Rates = $0.70 per minute
Time for Desired Lead = $14 * $3.50 per minute
Time for Desired Lead = 49 minutes

Work-5

A machine produces 35 bottles per minute, of which 3 are defective. How many minutes does the machine require to produce 8,000 nondefective bottles?

Canonical Solution

Number of Nondefective Bottles per Minute = 35 bottles per minute - 3 bottles per minute
Number of Nondefective Bottles per Minute = 32 bottles per minute
Time for Desired Nondefective Bottles = 8,000 bottles / 32 bottles per minute
Time for Desired Nondefective Bottles = 250 minutes

Goal-Specification Format Information

Givens
Total Number of Bottles per Minute =
Number of Defective Bottles per Minute =
Desired Number of Nondefective Bottles =

Unknowns
Number of Nondefective Bottles per Minute =
Time for Desired Nondefective Bottles =

Equation-Setup Format Information

Number of Nondefective Bottles per Minute =
Total Number of Bottles per Minute - Number of Defective Bottles per Minute
Time for Desired Nondefective Bottles =
Desired Number of Nondefective Bottles / Number of Nondefective Bottles per Minute

Faulty-Solution Format Information (using input rate for net rate: time = total bottles / avg rate => time = total bottles / input rate)

Time for Desired Nondefective Bottles = 8,000 bottles / 35 bottles per minute
Time for Desired Nondefective Bottles = 228.57 minutes

Work-6

If money is deposited into a soda machine at the rate of $12.00 per hour and is returned as change at the rate of $3.60 per hour, how many hours will it take the soda machine to accumulate $126.00?

Canonical Solution

Net Amount per Hour = $12.00 per hour - $3.60 per hour
Net Amount per Hour = $8.40 per hour
Time to Get Desired Amount of Money = $126.00 / $8.40 per hour
Time to Get Desired Amount of Money = 15 hours

Goal-Specification Format Information

Givens
Deposits per Hour =
Change per Hour =
Desired Amount of Money =

Unknowns
Net Amount per Hour =
Time to Get Desired Amount of Money =

Equation-Setup Format Information

Net Amount per Hour = Deposits per Hour - Change per Hour
Time to Get Desired Amount of Money = Desired Amount / Net Amount per Hour

Faulty-Solution Format Information (using output rate for net rate: time = total bottles / avg rate => time = total bottles / output rate)

Time to Get Desired Amount of Money = $126.00 / $3.60 per hour
Time to Get Desired Amount of Money = 35 hours

Work-7

If tickets for the evening performance of an event sell at the rate of 125 per hour while tickets for the matinee performance sell at the rate of 40 per hour, after how many hours do ticket sales for the evening performance exceed ticket sales for the matinee performance by 1,020 tickets?

Canonical Solution

Difference in Rate of Ticket Sales = 125 tickets per hour - 40 tickets per hour
Difference in Rate of Ticket Sales = 85 tickets per hour
Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Desired Amount =
    1,020 tickets / 85 tickets per hour
Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Desired Amount = 12 hours

Goal-Specification Format Information

Givens
    Rate of Ticket Sales for Evening Performance =
    Rate of Ticket Sales for Matinee Performance =
    Target Excess of Evening Tickets over Matinee Tickets =

Unknowns
    Difference in Rate of Ticket Sales =
    Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Target Amount =

Equation-Setup Format Information

Difference in Rate of Ticket Sales = Rate of Ticket Sales for Evening Performance
    - Rate of Ticket Sales for Matinee Performance
Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Desired Amount =
    Target Excess of Evening Tickets Over Matinee Tickets /
    Difference in Rate of Ticket Sales

Faulty-Solution Format Information (addition of separate times for net rates: net rate = evening sales rate - matinee sales rate => net rate = time for total sales at evening sales rate - time for total sales at matinee sales rate)

Difference in Rate of Ticket Sales =
    (1,020/125 tickets per hour) + (1,020/40 tickets per hour)
Difference in Rate of Ticket Sales = 8.16 tickets per hour + 25.5 tickets per hour
Difference in Rate of Ticket Sales = 33.66 tickets per hour
Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Desired Amount =
    1,020 tickets / 33.66 tickets per hour
Time for Evening Ticket Sales to Exceed Matinee Ticket Sales by Desired Amount = 30.30 hours

Work-8

If the Smith household uses 250 gallons of water per day whereas the Russell household uses 175 gallons per day, how many days elapse before the Smiths use 3,750 more gallons of water than the Russells?

Canonical Solution

Difference in Amount Used per Day = 250 galllons per day - 175 gallons per day
Difference in Amount Used per Day = 75 gallons per day
Time to Reach Target Difference = 3,750 gallons / 75 gallons per day
Time to Reach Target Difference = 50 days

Goal-Specification Format Information

Givens
    Amount Smiths Use per Day =
    Russells Use per Day =
    Target Difference in Use =
Unknowns
    Difference in Amount Used per Day =
    Time to Reach Target Difference =

Equation-Setup Format Information

Difference in Amount Used per Day = Amount Smiths Use per Day - Amount Russells Use per day
Time to Reach Target Difference = Target Difference in Use / Difference in Amount Used per Day

Faulty-Solution Format Information (input rate as total amount: time = total amount / avg rate => time = input rate amount / avg rate)

Difference in Amount Used per Day = 250 gallons per day - 175 gallons per day
Difference in Amount Used per Day = 75 gallons per day
Time to Reach Target Difference = 250 gallons / 75 gallons per day
Time to Reach Target Difference = 3 1/3 days

Interest-9

Money invested in a certain fund earns the same dividend each year, which is 5 percent of the original amount invested. In how many years will $750 invested in the fund earn total dividends equal to this amount invested?

Canonical Solutions

#1
Yearly Dividend = $750 * 5% per year
Yearly Dividend = $37.50 per year
Years Needed for Dividends to Equal Original Investment = $750 / $37.50 per year
Years Needed for Dividends to Equal Original Investment = 20 years

#2
100% Dividend = 5% Dividend per Year * X Years
X Years = 100% Dividend / 5% Dividend per Year
X Years = 20 Years

Goal-Specification Format Information

Givens
Original Investment =
Dividend Rate =
Target Total Dividend =

Unknowns
Yearly Dividend =
Years Needed for Dividends to Equal Target Total Dividend =

Equation-Setup Format Information

Yearly Dividend = Original Investment * Dividend Rate
Years Needed for Dividends to Equal Target Total Dividend =
Target Total Dividend / Yearly Dividend

Faulty-Solution Format Information (division error)

Yearly Dividend = $750 * 5% per year
Yearly Dividend = $37.50 per year
Years Needed for Dividends to Equal Original Investment = $750 / $37.50 per year
Years Needed for Dividends to Equal Original Investment = 11 years

Interest-10

On every $150 load of cement it delivers to a construction site, Acme Cement Company earns a 4 percent profit. How many loads must it deliver to the site to earn $150 in profit?

Canonical Solutions

#1
Profit per Load = $150 * 4% per load
Profit per Load = $6 per load
Loads Needed for Target Profit = $150 / $6 per load
Loads Needed for Target Profit = 25 loads

#2
100% Profit = 4% Profit per Load * X Loads
X Loads = 100% Profit / 4% Profit per Load
X Loads = 25 Loads

Goal-Specification Format Information

Givens
Cost per Load =
Percent Profit =
Target Profit =

Unknowns
Profit per Load =
Loads Needed for Target Profit =

Equation-Setup Format Information

Profit per Load = Cost per Load * Percent Profit
Loads Needed for Target Profit = Target Profit / Profit per Load

Faulty-Solution Format Information (treat percent as profit: loads needed = profit / profit rate => loads needed = profit / percent profit)

Loads Needed for Target Profit = $150 / 4%
Loads Needed for Target Profit = 3,750 loads

Interest-11

A graphics designer earns 2% of a $1500 yearly bonus for each shift of overtime she works. How many shifts of overtime must she work to earn the equivalent of the entire yearly bonus?

Canonical Solutions

#1
Amount Earned for Each Overtime Shift = $1500 * 2% per shift
Amount Earned for Each Overtime Shift = $30 per shift
Number of Shifts for Yearly Bonus = $1500 / $30 per shift
Number of Shifts for Yearly Bonus = 50 shifts

#2
100% of Amount Earned = 2% of Amount Earned per Shift * X Shifts
X Shifts = 100% of Amount Earned / 2% of Amount Earned per Shift
X Shifts = 50 Shifts

Goal-Specification Format Information

Givens
Yearly Bonus =
Percent of Bonus per Shift =
Target Bonus =
Unknowns
Amount Earned for Each Overtime Shift =
Number of Shifts for Target Bonus =

Equation-Setup Format Information

Amount Earned for Each Overtime Shift = Yearly Bonus * Percent of Bonus per Shift
Number of Shifts for Target Bonus = Target Bonus / Amount Earned for Each Overtime Shift

Faulty-Solution Format Information (substitute amount earned per shift for number of shifts needed)

Number of Shifts for Yearly Bonus = $1,500 * 2% per shift
Number of Shifts for Yearly Bonus = 30 shifts

Interest-12

The active ingredient is 0.25 percent of a 3-ounce dose of a certain cold remedy. What is the number of doses a patient must take before receiving 3 ounces of the active ingredient?

Canonical Solutions

#1
Ounces of Active Ingredient per Dose = 0.25 Percent per Dose * 3 ounces
Ounces of Active Ingredient per Dose = 0.0075 Ounces per Dose
Number of Doses Required = 3 Ounces / 0.0075 Ounces per Dose
Number of Doses Required = 400 Doses

#2
100% Active Ingredient = 0.25% Active Ingredient per Dose * X Doses
X Doses = 100% Active Ingredient / 0.25% Active Ingredient per Dose
X Doses = 400 Doses

Goal-Specification Format Information

Givens
    Number of Ounces in a Dose =
    Percent of Active Ingredient per Dose =
    Desired Amount of Active Ingredient =
Unknowns
    Ounces of Active Ingredient per Dose =
    Number of Doses Required =

Equation-Setup Format Information

Ounces of Active Ingredient per Dose =
    Percent of Active Ingredient per Dose * Number of Ounces in a Dose
Number of Doses Required =
    Desired Amount of Active Ingredient / Ounces of Active Ingredient per Dose

Faulty-Solution Format Information (treat percent as decimal)

Ounces of Active Ingredient per Dose = 0.25 percent * 3 ounces per dose
Ounces of Active Ingredient per Dose = 0.75 ounces per dose
Number of Doses Required = 3 ounces / 0.75 ounces per dose
Number of Doses Required = 4 doses

Interest-13

A sales representative earns a 4 percent commission on the sale of each encyclopedia set. How many $800 encyclopedia sets must the representative sell to earn a total commission equal to the price of one $800 encyclopedia set?

Canonical Solutions

#1
Commission per Set = 4 Percent per Set * $800
Commission per Set = $32 per Set
Sets for Target Commission = $800 / $32 per Set
Sets for Target Commission = 25 Sets

#2
100% Commission = 4% Commission per Set * X Sets
X Sets = 100% Commission / 4% Commission per Set
X Sets = 25 Sets

Goal-Specification Format Information

Givens
Cost of Each Set =
Percent Commission per Sale =
Target Commission =

Unknowns
Commission per Sale =
Sets for Target Commission =

Equation-Setup Format Information

Commission per Sale = Percent Commission per Sale * Cost of Each Set
Sets for Target Commission = Target Commission / Commission per Sale

Faulty-Solution Format Information (missing goal: report commission instead of number of sets)

Commission per Sale = 4 percent * $800
Commission per Sale = $32

Interest-14

An oil tanker filled to capacity with 10 million barrels of oil is leaking 0.02 percent of its capacity of oil each hour. At this rate, how many hours would it take for all of the tanker's oil to leak out?

Canonical Solutions

#1
Amount Leaked per Hour = 0.02 Percent per Hour * 10 Million Barrels
Amount Leaked per Hour = 0.002 Million Barrels per Hour
Time for Total Leakage = 10 Million Barrels / 0.002 Million Barrels per Hour
Time for Total Leakage = 5,000 Hours

#2
100% Leaked = 0.02% Leaked per Hour * X Hours
X Hours = 100% Leaked / 0.02% Leaked per Hour
X Hours = 5,000 Hours

Goal-Specification Format Information

Givens
Total Amount of Oil =
% Leaked per Hour =
Total Leakage =

Unknowns
Amount Leaked per Hour =
Time for Total Leakage =

Equation-Setup Format Information

Amount Leaked per Hour = % Leaked per Hour * Total Amount of Oil
Time for Total Leakage = Total Leakage / Amount Leaked per Hour

Faulty-Solution Format Information (decimal error: .0002 * 10 =.0002)

Amount Leaked per Hour = 0.02 percent * 10 million barrels
Amount Leaked per Hour = 0.0002 million barrels
Time for Total Leakage = 10 million barrels / 0.0002 million barrels per hour
Time for Total Leakage = 50,000 hours

Interest-15

Geologists estimate that 0.05 percent of a 2-mile-wide reef is eroding each century. At this rate, in how many centuries will the entire reef be eroded?

Canonical Solutions

#1
Erosion per Century = 0.05 Percent per Century * 2 Miles
Erosion per Century = 0.001 Miles per Century
Time for Complete Erosion = 2 miles / 0.001 Miles per Century
Time for Complete Erosion = 2,000 Centuries

#2
100% Erosion = 0.05% Erosion per Century * X Centuries
X Centuries = 100% Erosion / 0.05% Erosion per Century
X Centuries = 2,000 Centuries

Goal-Specification Format Information

Givens
Size of Reef =
Percent Eroding per Century =
Amount of Complete Erosion =

Unknowns
Erosion per Century =
Time for Complete Erosion =

Equation-Setup Format Information

Erosion per Century = Percent Eroding per Century * Size of Reef
Time for Complete Erosion = Amount of Complete Erosion / Erosion per Century

Faulty-Solution Format Information (percent treated as decimal AND multiplication for division: erosion amount = distance * erosion rate => erosion amount = distance / erosion rate)

Erosion per Century = 2 miles / 0.05 percent per century
Erosion per Century = 40 miles per century
Time for Complete Erosion = 2 miles / 40 miles per century
Time for Complete Erosion = 0.05 centuries

Interest-16

In one day, one laborer can pick 0.08 percent of the 900 bushels of fruit that an orchard is expected to yield, which is a "laborer-day" amount picked. How many laborer days would be required to pick all of the fruit that the orchard is expected to yield?

Canonical Solutions

#1
Bushels Picked in a Laborer Day = 0.08 Percent per Day * 900 Bushels
Bushels Picked in a Laborer Day = 0.72 Bushels per Laborer Day
Laborer Days Needed to Pick Targeted Number of Bushels =
900 Bushels / 0.72 Bushels per Laborer Day
Laborer Days Needed to Pick Targeted Number of Bushels = 1,250 Laborer Days

#2
100% of Bushels = 0.08% of Bushels per Laborer Day * X Laborer Days
X Laborer Days = 100% of Bushels / 0.08% of Bushels per Laborer Day
X Laborer Days = 1,250 Laborer Days

Goal-Specification Format Information

Givens
Total Number of Bushels =
Percent of Bushels Picked in a Laborer Day =
Targeted Number of Bushels to be Picked =

Unknowns
Bushels Picked in a Laborer Day =
Laborer Days Needed to Pick Targeted Number of Bushels =

Equation-Setup Format Information

Bushels Picked in a Laborer Day =
Percent of Bushels Picked in a Laborer Day * Total Number of Bushels
Laborer Days Needed to Pick Targeted Number of Bushels =
Targeted Number of Bushels to be Picked / Bushels Picked in a Laborer Day

Faulty-Solution Format Information (multiplication for division: days needed = amount picked / picking rate => days needed = amount picked * picking rate)

Bushels Picked in a Laborer Day = 0.08 percent * 900 bushels
Bushels Picked in a Laborer Day = 0.72 bushels per laborer day
Laborer Days Needed to Pick Targeted Number of Bushels =
900 bushels * 0.72 bushels per laborer day
Laborer Days Needed to Pick Targeted Number of Bushels = 648 laborer day

Distance = Rate x Time-17

On a 600-hundred mile motor trip, Bill averaged 45 miles per hour for the first 285 miles and 50 miles per hour for the remainder of the trip. If he started at 7:00 a.m. and drove nonstop, at what time, to the nearest minute, did he finish the trip?

Canonical Solution

Time for First Part of Trip = 285 miles / 45 mph
Time for First Part of Trip = 6 1/3 hours
Distance of Second Part of Trip = 600 miles - 285 miles
Distance of Second Part of Trip = 315 miles
Time for Second Part of Trip = 315 miles / 50 mph
Time for Second Part of Trip = 6 3/10 hours
Time for Total Trip = 6 1/3 hours + 6 3/10 hours
Time for Total Trip = 12 19/30 hours
Time for Total Trip = 12 hours 38 minutes
Ending Time of Trip = 7:00 a.m. + 12 hours 38 minutes
Ending Time of Trip = 7:38 p.m.

Goal-Specification Format Information

Givens
- Total Distance of Trip =
- Speed During First Part of Trip =
- Distance of First Part of Trip =
- Speed During Second Part of Trip =
- Starting Time of Trip =

Unknowns
- Time for First Part of Trip =
- Distance of Second Part of Trip =
- Time for Second Part of Trip =
- Time for Total Trip =
- Ending Time of Trip =

Equation-Setup Format Information

Time for First Part of Trip =
    Distance of First Part of Trip / Speed During First Part of Trip
Distance of Second Part of Trip = Total Distance of Trip - Distance of First Part of Trip
Time for Second Part of Trip =
    Distance of Second Part of Trip / Speed During Second Part of Trip
Time for Total Trip = Time for First Part of Trip + Time for Second Part of Trip
Ending Time of Trip = Starting Time of Trip + Time for Total Trip

Faulty-Solution Format Information (decimal hours as minutes: 12.63 hours => 12 hours 63 minutes)

Time for First Part of Trip = 285 miles / 45 mph
Time for First Part of Trip = 6.33 hours
Distance of Second Part of Trip = 600 miles - 285 miles
Distance of Second Part of Trip = 315 miles
Time for Second Part of Trip = 315 miles / 50 mph
Time for Second Part of Trip = 6.3 hours
Time for Total Trip = 6.33 hours + 6.3 hours
Time for Total Trip = 12.63 hours
Ending Time of Trip = 7:00 a.m. + 12 hours 63 minutes
Ending Time of Trip = 8:03 p.m.

Distance = Rate x Time-18

Workers started filling a 2,400 gallon tank through a single hose at 5:30 a.m. The first 800 gallons flowed into the tank at the rate of 75 gallons per hour through a twisted hose. Then, the twist was discovered and eliminated and the rest of the tank is being filled at the rate of 250 gallons per hour. At what time, to the nearest minute, will the filling of the tank be finished?

Canonical Solution

Filling Time 1 = 800 gallons / 75 gallons per hour
Filling Time 1 = 10.67 hours
Filling Amount 2 = 2,400 gallons - 800 gallons
Filling Amount 2= 1,600 gallons
Filling Time 2 = 1,600 gallons / 250 gallons per hour
Filling Time 2 = 6.4 hours
Total Filling Time = 10.67 hours + 6.4 hours
Total Filling Time = 17.07 hours
Total Filling Time = 17 hours 4 minutes
Ending Time for Filling = 5:30 a.m. + 17 hours 4 minutes
Ending Time for Filling = 10:34 p.m.

Goal-Specification Format Information

Givens
Tank Capacity =
Filling Rate 1 =
Filling Amount 1 =
Filling Rate 2 =
Starting Time for Filling =
Unknowns
Filling Time 1 =
Filling Amount 2 =
Filling Time 2 =
Total Filling Time =
Ending Time for Filling =

Equation-Setup Format Information

Filling Time 1 = Filling Amount 1 / Filling Rate 1
Filling Amount 2 = Tank Capacity - Filing Amount 1
Filling Time 2 = Filling Amount 2 / Filling Rate 2
Total Filling Time = Filling Time 1 + Filling Time 2
Ending Time for Filling = Starting Time for Filling + Total Filling Time

Faulty-Solution Format Information (a.m. for p.m.)

Filling Time 1 = 800 gallons / 75 gallons per hour
Filling Time 1 = 10.67 hours
Filling Amount 2 = 2,400 gallons - 800 gallons
Filling Amount 2= 1,600 gallons
Filling Time 2 = 1,600 gallons / 250 gallons per hour
Filling Time 2 = 6.4 hours
Total Filling Time = 10.67 hours + 6.4 hours
Total Filling Time = 17.07 hours
Total Filling Time = 17 hours 4 minutes
Ending Time for Filling = 5:30 a.m. + 17 hours 4 minutes
Ending Time for Filling = 10:34 a.m.

BEST COPY AVAILABLE

Distance = Rate x Time-19

A secretary typed the first 1,960 characters of a 6,424-character report at the rate of 105 characters per minute and is typing the rest of the report at the rate of 90 characters per minute. If the secretary began typing the report at 9:30 a.m. and types without interruption, at what time, to the nearest minute, will the secretary finish?

Canonical Solution

Typing Time for First Set = 1,960 characters / 105 characters per minute
Typing Time for First Set = 18.67 minutes
Number of Characters in Second Set = 6,424 characters - 1,960 characters
Number of Characters in Second Set = 4,464 characters
Typing Time for Second Set = 4,464 characters / 90 characters per minute
Typing Time for Second Set = 49.6 minutes
Total Typing Time = 18.67 minutes + 49.6 minutes
Total Typing Time = 68.27 minutes
Ending Time for Typing = 9:30 a.m. + 1 hour 8 minutes
Ending Time for Typing = 10:38 a.m.

Goal-Specification Format Information

Givens
- Total Characters =
- Typing Rate for First Set =
- Number of Characters in First Set =
- Typing Rate for Second Set =
- Starting Time for Typing =

Unknowns
- Typing Time for First Set =
- Number of Characters in Second Set =
- Typing Time for Second Set =
- Total Typing Time =
- Ending Time for Typing =

Equation-Setup Format Information

Typing Time for First Set = Number of Characters in First Set / Typing Rate for First Set
Number of Characters in Second Set = Total Characters - Number of Characters in First Set
Typing Time for Second Set =
    Number of Characters in Second Set / Typing Rate for Second Set
Total Typing Time = Typing Time for First Set + Typing Time for Second Set
Ending Time for Typing = Starting Time for Typing + Total Typing Time

Faulty-Solution Format Information (use first typing amount for second typing amount: number of characters in second set => number of characters in first set)

Typing Time for First Set = 1,960 characters / 105 characters per minute
Typing Time for First Set = 18.67 minutes
Typing Time for Second Set = 1,960 characters / 90 characters per minute
Typing Time for Second Set = 21.78 minutes
Total Typing Time = 18.67 minutes + 21.78 minutes
Total Typing Time = 40 minutes
Ending Time for Typing = 9:30 a.m. + 40 minutes
Ending Time for Typing = 10:10 a.m.

Distance = Rate x Time-20

A certain snowplow cleared the 10.4 mile straight part of a 16.7 mile highway at the rate of 3.2 miles per hour and cleared the remaining winding portion of the highway at the rate of 2.7 miles per hour. If the snowplow started at 11:15 a.m. and ran continuously, at what time, to the nearest minute, did it finish clearing the highway?

Canonical Solution

Time for Plowing Straight Part = 10.4 miles / 3.2 mph
Time for Plowing Straight Part = 3.25 hours
Distance of Winding Part = 16.7 miles - 10.4 miles
Distance of Winding Part = 6.3 miles
Time for Plowing Winding Part = 6.3 miles / 2.7 mph
Time for Plowing Winding Part = 2.33 hours
Time for Total Plowing = 2.33 hours + 3.25 hours
Time for Total Plowing = 5 hours 35 minutes
Ending Time of Plowing = 11:15 a.m. + 5 hours 35 minutes
Ending Time of Plowing = 4:50 p.m.

Goal-Specification Format Information

Givens
- Length of Highway =
- Rate of Plowing Straight Part =
- Distance of Straight Part =
- Rate of Plowing Winding Part =
- Starting Time of Plowing =

Unknowns
- Time for Plowing Straight Part =
- Distance of Winding Part =
- Time for Plowing Winding Part =
- Time for Total Plowing =
- Ending Time of Plowing =

Equation-Setup Format Information

Time for Plowing Straight Part = Distance of Straight Part / Rate of Plowing Straight Part
Distance of Winding Part = Length of Highway - Distance of Straight Part
Time for Plowing Winding Part = Distance of Winding Part / Rate of Plowing Winding Part
Time for Total Plowing = Time for Plowing Straight Part + Time for Plowing Winding Part
Ending Time of Plowing = Starting Time of Plowing + Time for Total Plowing

Faulty-Solution Format Information (use single unweighted rate for whole distance instead of one rate for each partial distance)

Plowing Speed = (3.2 mph + 2.7 mph) / 2
Plowing Speed = 2.95 mph
Time for Total Plowing = 16.7 miles / 2.95 mph
Time for Total Plowing = 5.66 hours
Ending Time of Plowing = 11:15 a.m. + 5 hours 40 minutes
Ending Time of Plowing = 4:55 p.m.

Distance = Rate x Time-21

A machine was set to produce the first 128 electrical components of the 420 needed for a certain production order at a rate of 24 components per hour and the rest of the order at a rate of 40 components per hour. If the machine started filling the order at 7:20 a.m. and ran continuously, at what time, to the nearest minute, was the order completed?

Canonical Solution

Product Time 1 = 128 components / 24 components per hour
Product Time 1 = 5.33 hours
Second Quantity = 420 components - 128 components
Second Quantity = 292 components
Production Time 2 = 292 components / 40 components per hour
Production Time 2 = 7.3 hours
Total Product Time = 5.33 hours + 7.3 hours
Total Product Time = 12 hours 38 minutes
Ending Time for Production = 7:20 a.m. + 12 hours 38 minutes
Ending Time for Production = 7:58 p.m.

Goal-Specification Format Information

Givens
- Total Components =
- Rate 1 =
- First Quantity =
- Rate 2 =
- Start Time =

Unknowns
- Product Time 1 =
- Second Quantity =
- Product Time 2 =
- Total Product Time =
- Ending Time for Production =

Equation-Setup Format Information

Product Time 1 = First Quantity / Rate 1
Second Quantity = Total Components - First Quantity
Product Time 2 = Second Quantity / Rate 2
Total Product Time = Product Time 1 + Product Time 2
Ending Time for Product = Start Time + Total Product Time

Faulty-Solution Format Information (.3 = 1/3)

Product Time 1 = 128 components / 24 components per hour
Product Time 1 = 5 1/3 hours
Second Quantity = 420 components - 128 components
Second Quantity = 292 components
Product Time 2 = 292 components / 40 components per hour
Product Time 2 = 7 1/3 hours
Total Product Time = 5 1/3 hours + 7 1/3 hours
Total Product Time = 12 hours 40 minutes
Ending Time for Production = 7:20 a.m. + 12 hours 40 minutes
Ending Time for Production = 8:00 p.m.

Distance = Rate x Time-22

A grocery store sold 317 cans of its brand of tomato sauce yesterday. The first 62 cans were sold at the regular price at the rate of 12 cans per hour. The rest of the cans, which had been marked in error at a lower price, were sold at the rate of 34 cans per hour. If the tomato sauce began selling at 8:00 a.m., at what time, to the nearest minute, was the last can sold?

Canonical Solution

Time to Sell Cans at Regular Price = 62 cans / 12 cans per hour
Time to Sell Cans at Regular Price = 5.17 hours
Cans Sold at Lower Price = 317 cans - 62 cans
Cans Sold at Lower Price = 255 cans
Time to Sell Cans at Lower Price = 255 cans / 34 cans per hour
Time to Sell Cans at Lower Price = 7.5 hours
Total Sales Time = 5.17 hours + 7.5 hours
Total Sales Time = 12 hours 40 minutes
Ending Time for Sales = 8:00 a.m. + 12 hours 40 minutes
Ending Time for Sales = 8:40 p.m.

Goal-Specification Format Information

Givens
Total Cans =
Sales Rate at Regular Price =
Cans Sold at Regular Price =
Sales Rate at Lower Price =
Starting Time for Sales =

Unknowns
Time to Sell Cans at Regular Price =
Cans Sold at Lower Price =
Time to Sell Cans at Lower Price =
Total Sales Time =
Ending Time for Sales =

Equation-Setup Format Information

Time to Sell Cans at Regular Price =
Cans Sold at Regular Price / Sales Rate at Regular Price
Cans Sold at Lower Price = Total Cans - Cans Sold at Regular Price
Time to Sell Cans at Lower Price = Cans Sold at Lower Price / Sales Rate at Lower Price
Total Sales Time = Time to Sell Cans at Regular Price + Time to Sell Cans at Lower Price
Ending Time for Sales = Starting Time for Sales + Total Sales Time

Faulty-Solution Format Information (invert time formula: time for sales = number of cans / rate of sales => time for sales = rate of sales / number of cans)

Time to Sell Cans at Regular Price = 12 cans per hour / 62 cans
Time to Sell Cans at Regular Price = 0.19 hours
Cans Sold at Lower Price = 317 cans - 62 cans
Cans Sold at Lower Price = 255 cans
Time to Sell Cans at Lower Price = 34 cans per hour / 255 cans
Time to Sell Cans at Lower Price = 0.13 hours
Total Sales Time = 0.19 hours + 0.13 hours
Total Sales Time = 19 minutes
Ending Time for Sales = 8:00 a.m. + 19 minutes
Ending Time for Sales = 8:19 a.m.

Distance = Rate x Time-23

A fire is burning in a forest of several thousand acres. Firefighters are attempting to confine the fire to one region covering 705 acres, and plan to light a backfire at the moment the fire reaches the edge of that region, after having burned the entire region. The first 459 acres burned at a rate of 90 acres per hour. The wind then diminished, and the fire is now burning at the rate of 36 acres per hour. If the fire started at 10 a.m. and wind conditions do not change, at what time, to the nearest minute, should the backfire be lit?

Canonical Solution

Burning Time 1 = 459 acres / 90 acres per hour
Burning Time 1 = 5.1 hours
Acres Amount 2 = 705 acres - 459 acres
Acres Amount 2 = 246 acres
Burning Time 2 = 246 acres / 36 acres per hour
Burning Time 2 = 6.83 hours
Total Burning Time = 5.1 hours + 6.83 hours
Total Burning Time = 11 hours 56 minutes
Time Backfire Should Be Lit = 10 a.m. + 11 hours 56 minutes
Time Backfire Should Be Lit = 9:56 p.m.

Goal-Specification Format Information

Givens
Total Acres =
Burning Rate 1 =
Acres Amount 1 =
Burning Rate 2 =
Starting Time for Burning =
Unknowns
Burning Time 1 =
Acres Amount 2 =
Burning Time 2 =
Total Burning Time =
Time Backfire Should Be Lit =

Equation-Setup Format Information

Burning Time 1 = Acres Amount 1 / Burning Time 1
Acres Amount 2 = Total Acres - Acres Amount 1
Burning Time 2 = Acres Amount 2 / Burning Time 2
Total Burning Time = Burning Time 1 + Burning Time 2
Time Backfire Should Be Lit = Starting Time for Burning + Total Burning Time

Faulty-Solution Format Information (treat numerator and denominator as independent in fraction addition: 1/10 + 5/6 = 6/16)

Burning Time 1 = 459 acres / 90 acres per hour
Burning Time 1 = 5 1/10 hours
Acres Amount 2 = 705 acres - 459 acres
Acres Amount 2 = 246 acres
Burning Time 2 = 246 acres / 36 acres per hour
Burning Time 2 = 6 5/6 hours
Total Burning Time = 5 1/10 hours + 6 5/6 hours
Total Burning Time = 11 6/16 hours
Total Burning Time = 11 hours 22 minutes
Time Backfire Should Be Lit = 10 a.m. + 11 hours 22 minutes
Time Backfire Should Be Lit = 9:22 p.m.

Distance = Rate x Time-24

During a break in its program at 6:15 p.m., a TV station began a fund-raising campaign that has a goal of raising $13,900 through viewer solicitation. It raised the first $7,000 at the rate of $1,200 per minute. When a special appeal began, pledges came in at the rate of $3,000 per minute. If funds continue to come in at the same rate as during the special appeal, until what time, to the nearest minute, must the program break continue in order for the station to reach its goal?

Canonical Solution

Time During Initial Appeal = $7,000 / $1,200 per minute
Time During Initial Appeal = 5.83 minutes
Amount During Special Appeal = $13,900 - $7,000
Amount During Special Appeal = $6,900
Time After Special Appeal = $6,900 / $3,000 per minute
Time After Special Appeal = 2.3 minutes
Total Program Time Needed = 5.83 minutes + 2.3 minutes
Total Program Time Needed = 8 minutes
End Time of Program = 6:15 p.m. + 8 minutes
End Time of Program = 6:23 p.m.

Goal-Specification Format Information

Givens
- Start Time of Program =
- Total Goal =
- Amount From Initial Appeal =
- Initial Rate of Pledges =
- Rate After Special Appeal =

Unknowns
- Time During Initial Appeal =
- Amount During Special Appeal =
- Time After Special Appeal =
- Total Program Time Needed =
- End Time of Program =

Equation-Setup Format Information

Time During Initial Appeal = Amount From Initial Appeal / Initial Rate of Pledges
Amount During Special Appeal = Total Goal - Amount From Initial Appeal
Time After Special Appeal = Amount During Special Appeal / Rate After Special Appeal
Total Program Time Needed = Time During Initial Appeal + Time After Special Appeal
End Time of Program = Start Time of Program + Total Program Time Needed

Faulty-Solution Format Information (missing final goal: report total time needed instead of ending time)

Time During Initial Appeal = $7,000 / $1,200 per minute
Time During Initial Appeal = 5.83 minutes
Amount During Special Appeal = $13,900 - $7,000
Amount During Special Appeal = $6,900
Time After Special Appeal = $6,900 / $3,000 per minute
Time After Special Appeal = 2.3 minutes
Total Program Time Needed = 5.83 minutes + 2.3 minutes
Total Program Time Needed = 8 minutes

Graduated Rate-25

A lawyer charges $100 for the first hour of service and $75 for each additional hour. A bill of $625 represents how many hours of the lawyer's service?

Canonical Solution

Lawyer's Charge After First Hour = $625 - $100
Lawyer's Charge After First Hour = $525
Number of Additional Hours = $525 / $75
Number of Additional Hours = 7 hours
Total Number of Hours = 7 hours + 1 hour
Total Number of Hours = 8 hours

Goal-Specification Format Information

Givens
Charge for First Hour =
Rate for Additional Hours =
Total Charge for Lawyer's Service =

Unknowns
Lawyer's Charge After First Hour =
Number of Additional Hours =
Total Number of Hours =

Equation-Setup Format Information

Lawyer's Charge After First Hour =
Total Charge for Lawyer's Service - Charge for First Hour
Number of Additional Hours = Lawyer's Charge After First Hour / Rate for Additional Hours
Total Number of Hours = First Hour + Number of Additional Hours

Faulty-Solution Format Information (switch rate for first hour and rate for subsequent hours)

Lawyer's Charge After First Hour = $625 - $75
Lawyer's Charge After First Hour = $550
Number of Additional Hours = $550 / $100
Number of Additional Hours = 5.5 hours
Total Number of Hours = 5.5 hours + 1 hour
Total Number of Hours = 6 hours 30 minutes

Graduated Rate-26

A taxi charges $2.00 for the first mile and $1.40 for each additional mile. If the total charge is $18.80 for a certain trip, how many miles long is the trip?

Canonical Solution

Added Miles Cost = $18.80 - $2.00
Added Miles Cost = $16.80
Miles After First Mile = $16.80 / $1.40
Miles After First Mile = 12 miles
Total Miles for Trip = 1 mile + 12 miles
Total Miles for Trip = 13 miles

Goal-Specification Format Information

Givens
Cost for First Mile =
Added Miles Rate =
Total Cost =

Unknowns
Added Miles Cost =
Miles After First Mile =
Total Miles for Trip =

Equation-Setup Format Information

Added Miles Cost = Total Cost - Cost for First Mile
Miles After First Mile of Trip = Added Miles Cost / Added Miles Rate
Total Miles for Trip = First Mile + Miles After First Mile

Faulty-Solution Format Information (drop charge for initial mile)

Added Miles Cost = $18.80 - $2.00
Added Miles Cost = $16.80
Miles After First Mile = $16.80 / $1.40
Miles After First Mile = 12 miles
Total Miles for Trip = 12 miles

Graduated Rate-27

In year X, postage rates for first-class letters were $0.25 for the first ounce and $0.20 for each additional ounce. In year X, if the total postage due for a certain first-class letter was $2.25, how many ounces did the letter weigh?

Canonical Solution

Postage Charge After First Ounce = $2.25 - $0.25
Postage Charge After First Ounce = $2.00
Number of Ounces After First Ounce = $2.00 / $0.20
Number of Ounces After First Ounce = 10 ounces
Total Number of Ounces = 10 ounces + 1 ounce
Total Number of Ounces = 11 ounces

Goal-Specification Format Information

Givens
Charge for First Ounce =
Rate for Additional Ounces =
Total Charge for Letter =

Unknowns
Postage Charge After First Ounce =
Number of Ounces After First Ounce =
Total Number of Ounces =

Equation-Setup Format Information

Postage Charge After First Ounce = Total Charge for Letter - Charge for First Ounce
Number of Ounces After First Ounce =
Postage Charge After First Ounce / Rate for Additional Ounces
Total Number of Ounces = First Ounce + Number of Additional Ounces

Faulty-Solution Format Information (switch rate for first ounce and rate for subsequent ounces)

Postage Charge After First Ounce = $2.25 - $0.20
Postage Charge After First Ounce = $2.05
Number of Ounces After First Ounce = $2.05 / $0.25
Number of Ounces After First Ounce = 8.2
Total Number of Ounces = 8.2 ounces + 1 ounce
Total Number of Ounces = 9.2 ounces

Graduated Rate-28

Telephone charges for evening calls from Raytown to Cherryville are $0.42 for the first minute and $0.12 for each additional minute. If the total charge for such a call is $2.70, how many minutes long is the call?

Canonical Solution

Phone Charge After First Minute = $2.70 - $0.42
Phone Charge After First Minute = $2.28
Number of Additional Minutes = $2.28 / $0.12
Number of Additional Minutes = 19 minutes
Total Number of Minutes = 19 minutes + 1 minute
Total Number of Minutes = 20 minutes

Goal-Specification Format Information

Givens
Charge for First Minute =
Rate for Additional Minutes =
Total Charge for Call =

Unknowns
Phone Charge After First Minute =
Number of Additional Minutes =
Total Number of Minutes =

Equation-Setup Format Information

Phone Charge After First Minute = Total Charge for Call - Charge for First Minute
Number of Additional Minutes =
Phone Charge After First Minute / Rate for Additional Minutes
Total Number of Minutes = First Minute + Number of Additional Minutes

Faulty-Solution Format Information (use single unweighted rate for entire call instead of one rate for first minute and a second rate for remainder of call)

Average Phone Rate = ($0.42 + $0.12) /2
Average Phone Rate = $0.27
Total Number of Minutes = $2.70 / $0.27
Total Number of Minutes = 10 minutes

Graduated Rate-29

The charge to rent a boat is $20 for the first hour and $12 for each additional hour. If the total charge to rent a boat was $116, for how many hours was the boat rented?

Canonical Solution

Rental Charge After First Hour = $116 - $20
Rental Charge After First Hour = $96
Number of Rental Hours After First Hour = $96 / $12
Number of Rental Hours After First Hour = 8 hours
Total Number of Hours = 8 hours + 1 hour
Total Number of Hours = 9 hours

Goal-Specification Format Information

Givens
Rental Charge for First Hour =
Rate for Additional Hours =
Total Charge for Rental =

Unknowns
Rental Charge after First Hour =
Number of Rental Hours After First Hour =
Total Number of Hours =

Equation-Setup Format Information

Rental Charge After First Hour = Total Charge for Rental - Rental Charge for First Hour
Number of Rental Hours After First Hour =
Rental Charge After First Hour / Rate for Additional Hours
Total Number of Hours = First Hour + Number of Rental Hours After First Hour

Faulty-Solution Format Information (assume single rental rate and ignore first hour rate: total time = total cost / rate after first hour)

Total Number of Hours = $116 / $12
Total Number of Hours = 9.67 hours

Graduated Rate-30

An entertainer's contract specifies payment of $25,000 for the first performance and $18,000 for each additional performance. If the contract specifies a total payment of $277,000 for the entertainer's performances, how many performances are required?

Canonical Solution

Total Charge After First Performance = $277,000 - $25,000
Total Charge After First Performance = $252,000
Number of Additional Performances = $252,000 / $18,000
Number of Additional Performances = 14
Total Number of Performances = 14 performances + 1 performance
Total Number of Performances = 15 performances

Goal-Specification Format Information

Givens
Charge for First Performance =
Rate for Additional Performances =
Total Charge for Contract =

Unknowns
Total Charge After First Performance =
Number of Additional Performances =
Total Number of Performances =

Equation-Setup Format Information

Total Charge After First Performance = Total Charge for Contract - Charge for First Performance
Number of Additional Performances =
Contract Charge After First Performance / Rate for Additional Performances
Total Number of Performances = First Performance + Number of Additional Performances

Faulty-Solution Format Information (use first performance charge for subsequent performance charges)

Total Charge After First Performance = $277,000 - $25,000
Total Charge After First Performance = $252,000
Number of Additional Performances = $252,000 / $25,000
Number of Additional Performances = 10 performances
Total Number of Performances = 10 performances + 1 performance
Total Number of Performances = 11 performances

Graduated Rate-31

Parking charges at a certain lot are $1.25 for the first hour and $0.75 for each additional hour. If the total charge to park a car at the lot was $8.00, for how many hours was the car parked?

Canonical Solution

Parking Charge After First Hour = $8.00 - $1.25
Parking Charge After First Hour = $6.75
Number of Additional Hours = $6.75 / $0.75
Number of Additional Hours = 9 hours
Total Number of Hours = 9 hours + 1 hour
Total Number of Hours = 10 hours

Goal-Specification Format Information

Givens
Charge for First Hour =
Rate for Additional Hours =
Total Charge for Parking =

Unknowns
Parking Charge After First Hour =
Number of Additional Hours =
Total Number of Hours =

Equation-Setup Format Information

Parking Charge After First Hour = Total Charge for Parking - Charge for First Hour
Number of Additional Hours = Parking Charge After First Hour / Rate for Additional Hours
Total Number of Hours = First Hour + Number of Additional Hours

Faulty-Solution Format Information (add instead of subtract for charge after first hour: charge after first hour = total charge + charge for first hour)

Parking Charge After First Hour = $8.00 + $1.25
Parking Charge After First Hour = $9.25
Number of Additional Hours = $9.25 / $0.75
Number of Additional Hours = 12 1/3 hours
Total Number of Hours = 12 1/3 hours + 1 hour
Total Number of Hours = 13 hours 20 minutes

Graduated Rate-32

A hardware store charges $1.95 for the first widget purchased and $1.50 for each additional widget purchased. If the total charge for widgets was $22.95, how many widgets were purchased?

Canonical Solution

Charge for Additional Widgets = $22.95 - $1.95
Charge for Additional Widgets = $21.00
Number of Additional Widgets = $21.00 / $1.50
Number of Additional Widgets = 14 widgets
Total Number of Widgets = 14 widgets + 1 widget
Total Number of Widgets = 15 widgets

Goal-Specification Format Information

Givens
Charge for First Widget =
Rate for Additional Widgets =
Total Charge for Purchase =

Unknowns
Charge for Additional Widgets =
Number of Additional Widgets =
Total Number of Widgets =

Equation-Setup Format Information

Charge for Additional Widgets = Total Charge for Purchase - Charge for First Widget
Number of Additional Widgets = Charge for Additional Widgets / Rate for Additional Widgets
Total Number of Widgets = First Widget + Number of Additional Widgets

Faulty-Solution Format Information (use difference between rates as single rate for all widgets: number of widgets = total cost / difference between rates)

Rate Difference = $1.95 - $1.50
Rate Difference = $0.45
Total Number of Widgets = $22.95 / $0.45
Total Number of Widgets = 51 widgets

Distance = Rate x Time (2)-33

A car is traveling at an average speed of 80 kilometers per hour. On the average, how many seconds does it take the car to travel K kilometers?

Canonical Solutions

#1
Kilometers Traveled per Second = 80 Kilometers per Hour / 3,600 Seconds per Hour
Kilometers Traveled per Second = 1/45 Kilometers per Second
Seconds to Travel K Kilometers = K Kilometers / 1/45 Kilometer per Second
Seconds to Travel K Kilometers = 45K Seconds

#2
Hours to Travel K Kilometers = K Kilometers / 80 Kilometers per Hour
Hours to Travel K Kilometers = K/80 Hours
Seconds to Travel K Kilometers = K/80 Hours * 3,600 Seconds per Hour
Seconds to Travel K Kilometers = 45K Seconds

#3
Seconds per Kilometer = 1/80 Hour per Kilometer * 3,600 Seconds per Hour
Seconds per Kilometer = 45 Seconds per Kilometer
Seconds to Travel K Kilometers = 45 Seconds per Kilometer * K kilometers
Seconds to Travel K Kilometers = 45K Seconds

Goal-Specification Format Information

Givens
Kilometers Traveled per Hour =
Kilometers to Travel =

Unknowns
Kilometers Traveled per Second =
Seconds to Travel K Kilometers =

Equation-Setup Format Information

Kilometers Traveled per Second = Kilometers Traveled per Hour / Seconds per Hour
Seconds to Travel K Kilometers = Kilometers to Travel / Kilometers Traveled per Second

Faulty-Solution Format Information (convert to minutes instead of hours)

Kilometers Traveled per Second = 80 Kilometers per Hour / 60 Seconds per Hour
Kilometers Traveled per Second = 1 1/3 Kilometers per Second
Seconds to Travel K Kilometers = K Kilometers / 1 1/3 Kilometer per Second
Seconds to Travel K Kilometers = 3/4 K Seconds

Distance = Rate x Time(2)-34

An automated toll booth collects money from an average of 600 cars per hour. At that rate, how many seconds does it take that booth to collect money from C cars, on the average?

Canonical Solutions

#1
Cars Through Toll Booth per Second = 600 Cars per Hour / 3,600 Seconds per Hour
Cars Through Toll Booth per Second = 1/6 Cars per Second
Seconds to Collect Money from C Cars = C Cars / 1/6 Cars per Second
Seconds to Collect Money from C Cars = 6C Seconds

#2
Hours to Collect Money from C Cars = C Cars / 600 Cars per Hour
Hours to Collect Money from C Cars = C/600 Hours
Seconds to Collect Money from C Cars = C/600 Hours * 3,600 Seconds per Hour
Seconds to Collect Money from C Cars = 6C Seconds

#3
Seconds per Car Through Toll Booth = 1/600 Hour per Car * 3,600 Seconds per Hour
Seconds per Car Through Toll Booth = 6 Seconds per Car
Seconds to Collect Money from C Cars = 6 Seconds per Car * C Cars
Seconds to Collect Money from C Cars = 6C Seconds

Goal-Specification Format Information

Givens
Cars Through Toll Booth per Hour =
Number of Cars from Which to Collect Money =

Unknowns
Cars Through Toll Booth per Second =
Seconds to Collect Money from C Cars =

Equation-Setup Format Information

Cars Through Toll Booth per Second = Cars Through Toll Booth per Hour / Seconds per Hour
Seconds to Collect Money from C Cars =
Number of Cars from Which to Collect Money / Cars Through Toll Booth per Second

Faulty-Solution Format Information (assume the constant C = 1)

Cars Through Toll Booth per Second = 600 Cars per Hour / 3,600 Seconds per Hour
Cars Through Toll Booth per Second = 1/6 Cars per Second
Seconds to Collect Money from C Cars = 1 Car / 1/6 Cars per Second
Seconds to Collect Money from C Cars = 6 Seconds

Distance = Rate x Time(2)-35

Each hour, an average of 8,000 letters are sorted by a postal machine. At that rate, how many seconds does it take, on average, for the machine to sort L letters?

Canonical Solutions

#1
Letters Sorted per Second = 8,000 Letters per Hour / 3,600 Seconds per Hour
Letters Sorted per Second = 20/9 Letters per Second
Seconds to Sort L Letters = L Letters / 20/9 Letters per Second
Seconds to Sort L Letters = 9L/20 Seconds

#2
Hours to Sort L Letters = L Letters / 8,000 Letters per Hour
Hours to Sort L Letters = L/8000 Hours
Seconds to Sort L Letters = L/8000 Hours * 3,600 Seconds per Hour
Seconds to Sort L Letters = 9L/20 Seconds

#3
Seconds per Letter = 1/8000 Hours per Letter * 3,600 Seconds per Hour
Seconds per Letter = 9/20 Seconds per Letter
Seconds to Sort L Letters = 9/20 Seconds per Letter * L Letters
Seconds to Sort L Letters = 9L/20 Seconds

Goal-Specification Format Information

Givens
Letters Sorted per Hour =
Letters to Sort =

Unknowns
Letters Sorted per Second =
Seconds to Sort L Letters =

Equation-Setup Format Information

Letters Sorted per Second = Letters Sorted per Hour / Seconds per Hour
Seconds to Sort L Letters = Letters to Sort / Letters Sorted per Second

Faulty-Solution Format Information (multiply instead of divide for time: time to sort = number of letters / sorting rate => time to sort = number of letters * sorting rate)

Letters Sorted per Second = 8,000 Letters per Hour / 3,600 Seconds per Hour
Letters Sorted per Second = 20/9 Letters per Second
Seconds to Sort L Letters = L Letters * 20/9 Letters per Second
Seconds to Sort L Letters = 20L/9 Seconds

Distance = Rate x Time(2)-36

A grocery store sells an average of 48,000 apples in a year. At that rate, on the average, how many months does it take the store to sell A apples?

Canonical Solutions

#1
Apples Sold per Month = 48,000 Apples per Year / 12 Months per Year
Apples Sold per Month = 4,000 Apples per Month
Months to Sell A Apples = A Apples / 4,000 Apples per Month
Months to Sell A Apples = A /4,000 Months

#2
Years to Sell A Apples = A Apples / 48,000 Apples per Year
Years to Sell A Apples = A/48,000 Years
Months to Sell A Apples = A/48,000 Years * 12 Months per Year
Months to Sell A Apples = A/4,000 Months

#3
Months per Apple Sold = 1/48,000 Year per Apple * 12 Months per Year
Months per Apple Sold = 1/4,000 Months per Apple
Months to Sell A Apples = 1/4,000 Months per Apple * A Apples
Months to Sell A Apples = A/4,000 Months

Goal-Specification Format Information

Givens
Apples Sold per Year =
Number of Apples to Sell =

Unknowns
Apples Sold per Month =
Months to Sell A Apples =

Equation-Setup Format Information

Apples Sold per Month = Apples Sold per Year / Years per Month
Months to Sell A Apples = Number of Apples to Sell / Apples Sold per Month

Faulty-Solution Format Information (solve for wrong units: years instead of months)

Years to Sell A Apples = A Apples / 48,000 Apples per Year
Years to Sell A Apples = A / 48,000 Years

Distance = Rate x Time(2)-37

A car manufacturing plant produces an average of 875 cars per week. In order for that plant to manufacture L cars, how many days does would it take on the average?

Canonical Solutions

#1
Cars Produced per Day = 875 Cars per Week / 7 Days per Week
Cars Produced per Day = 125 Cars per Day
Days to Produce L Cars = L Cars / 125 Cars per Day
Days to Produce L Cars = L/125 Days

#2
Weeks to Produce L Cars = L Cars / 875 Cars per Week
Weeks to Produce L Cars = L/875 Weeks
Days to Produce L Cars = L/875 Weeks * 7 Days per Week
Days to Produce L Cars = L/125 Days

#3
Days per Car Produced = 1/875 Week per Car * 7 Days per Week
Days per Car Produced = 1/125 Days per Car
Days to Produce L Cars = 1/125 Days per Car * L Cars
Days to Produce L Cars = L/125 Days

Goal-Specification Format Information

Givens
Cars Produced per Week =
Number of Cars to Produce =

Unknowns
Cars Produced per Day =
Days to Produce L Cars =

Equation-Setup Format Information

Cars Produced per Day = Cars Produced per Week / Days per Week
Days to Produce L Cars = Number of Cars to Produce / Cars Produced per Day

Faulty-Solution Format Information (solve for rate instead of time: missing final goal)

Cars Produced per Day = 875 Cars per Week / 7 Days per Week
Cars Produced per Day = 125 Cars per Day

Distance = Rate x Time(2)-38

A taxicab driver earns on average $144 for each 8-hour shift. On average, how many hours does it take the driver to earn C dollars?

Canonical Solutions

#1
Dollars Earned per Hour = $144 per Shift / 8 Hours per Shift
Dollars Earned per Hour = $18 per Hour
Hours to Earn C Dollars = C Dollars / $18 per Hour
Hours to Earn C Dollars = C /18 Hours

#2
Number of Shifts to Earn C Dollars = C Dollars / $144 per Shift
Number of Shifts to Earn C Dollars = C/144 Shifts
Hours to Earn C Dollars = C/144 Shifts * 8 Hours per Shift
Hours to Earn C Dollars = C/18 Hours

#3
Hours per Dollar Earned = 1/144 Shifts per Dollar * 8 Hours per Shift
Hours per Dollar Earned = 1/18 Hours per Dollar
Hours to Earn C Dollars = 1/18 Hours per Dollar * C Dollars
Hours to Earn C Dollars = C/18 Hours

Goal-Specification Format Information

Givens
Dollars Earned per Shift =
Desired Earnings =

Unknowns
Dollars Earned per Hour =
Hours to Earn C Dollars =

Equation-Setup Format Information

Dollars Earned per Hour = Dollars Earned per Shift / Hours per Shift
Hours to Earn C Dollars = Desired Earnings / Dollars Earned per Hour

Faulty-Solution Format Information (invert time: time = earnings / earnings rate => time = earnings rate / earnings)

Dollars Earned per Hour = $144 per Day / 8 Hours per Day
Dollars Earned per Hour = $18 per Hour
Hours to Earn C Dollars = $18 per Hour / C Dollars
Hours to Earn C Dollars = 18/C Hours

Distance = Rate x Time(2)-39

A local phone system processes an average of 12,000 calls each hour. On the average, how many seconds would it take the phone system to process K calls?

Canonical Solutions

#1
Calls Processed per Second = 12,000 Calls Processed per Hour / 3,600 Seconds per Hour
Calls Processed per Second = 3 1/3 Calls Processed per Second
Seconds to Process K Calls = K Calls / 3 1/3 Calls Processed per Second
Seconds to Process K Calls = 3K/10 Seconds

#2
Hours to Process K Calls = K Calls / 12,000 Calls per Hour
Hours to Process K Calls = K/12,000 Hours
Seconds to Process K Calls = K/12,000 Hours * 3,600 Seconds per Hour
Seconds to Process K Calls = 3K/10 Seconds

#3
Seconds per Call = 1/12000 Hour per Call * 3,600 Seconds per Hour
Seconds per Call = 3/10 Seconds per Call
Seconds to Process K Calls = 3/10 Seconds per Call * K Calls
Seconds to Process K Calls = 3K/10 Seconds

Goal-Specification Format Information

Givens
Calls Processed per Hour =
Number of Calls to Process =

Unknowns
Calls Processed per Second =
Seconds to Process K Calls =

Equation-Setup Format Information

Calls Processed per Second = Calls Processed per Hour / Seconds per Hour
Seconds to Process K Calls = Number of Calls to Process / Calls Processed per Second

Faulty-Solution Format Information (ignore time conversion)

Time to Process K Calls = K Calls / 12,000
Time to Process K Calls = K/12,000 Seconds

Distance = Rate x Time(2)-40

A bank teller machine processes 1,600 transactions per hour. How many minutes, on average, does it take the teller machine to process A transactions?

Canonical Solutions

#1
Transactions Processed per Minute = 1,600 Transactions per Hour / 60 Minutes per Hour
Transactions Processed per Minute = 26 2/3 Transactions per Minute
Minutes to Process A Transactions = A Transactions / 26 2/3 Transactions per Minute
Minutes to Process A Transactions = 3A/80 Minutes

#2
Hours to Process A Transactions = A Transactions / 1,600 Transactions per Hour
Hours to Process A Transactions = A/1,600 Hours
Minutes to Process A Transactions = A/1,600 Hours * 60 Minutes per Hour
Minutes to Process A Transactions = 3A/80 Minutes

#3
Minutes per Transaction = 1/1,600 Hour per Transaction * 60 Minutes per Hour
Minutes per Transaction = 3/10 Minutes per Transaction
Minutes to Process A Transactions = 3/10 Minutes per Transaction * A Transactions
Minutes to Process A Transactions = 3A/80 Minutes

Goal-Specification Format Information

Givens
Transactions Processed per Hour =
Transactions to Process =

Unknowns
Transactions Processed per Minute =
Minutes to Process A Transactions =

Equation-Setup Format Information

Transactions Processed per Minute = Transactions Processed per Hour / Minutes per Hour
Minutes to Process A Transactions =
Transactions to Process / Transactions Processed per Minute

Faulty-Solution Format Information (invert time conversion: transactions per minute = transactions per hour / minutes per hour => transactions per minute = transactions per hour * minutes per hour)

Transactions Processed per Minute = 1,600 A Transactions per Hour / 60 Minutes per Hour
Transactions Processed per Minute = 26 2/3 Transactions per Minute
Minutes to Process A Transactions = 26 2/3 Transactions per Minute / A Transactions
Minutes to Process A Transactions = (26 2/3)/A Minutes

Appendix B

Scoring Rubric

From Machine-scorable complex constructed-response quantitative items: Agreement between expert system and human raters' scores (RR-91-11) by M. M. Sebrechts, R. E. Bennett, and D. A. Rock. (Princeton, NJ: Educational Testing Service, 1991).

1. If the student provides two or more solutions, consider only the best one. In general, do not deduct credit if the student explicitly corrects errors.

2. Consider all available information including that in the "Calculations Space."

3. If only the final answer is present and it is correct, give full credit because there is no process on which to make any other decision. In all other cases, the total score for the problem is the sum of the scores for each goal.

4. Each goal is worth 3 points. Deduct points as follows:

   a. Deduct 3 points if the goal is missing and is not implicitly satisfied. A goal is considered missing when there is no reasonable attempt to solve for it. A goal is considered to be implicitly satisfied if it can be inferred from other parts of the solution.

   b. Deduct 2 points if the goal is present but contains an uncorrected structural error (e.g., inverting the dividend and the divisor, confusing operators). For a goal to be considered present but structurally incorrect, it must be clearly evident that the student is making an attempt--however misguided--to solve the goal (thereby showing awareness that solving for that goal is a step in the problem's solution process). The minimal evidence needed to indicate such an attempt is the presence of a reasonable expression bound to a label that can be unambiguously associated with that goal.

   c. Deduct 1 point for each computational error within a present goal. Count as computational errors miscalculations (including those beyond the required level of precision), transcription errors (values incorrectly copied from one part of the problem to another), errors in copying a given from the problem statement, conversion errors (unless otherwise indicated), and, for the last goal only, failing to reduce the final answer to a single value. Only deduct for the same computational error once. For all computational errors, carry through the result to subsequent goals, giving full credit to those subsequent goals if they are structurally and computationally correct given their incorrect input.

   d. Deduct 1 point for failing to carry the result of a goal to the required level of precision (i.e., two decimal places or the precision required by the individual problem, whichever is greater).

   e. Deduct 0 points if the goal is present and correct. A goal should be considered to be present and correct if (1) the result and the method are correct, (2) the result is correct and the method is not identifiably faulty, or (3) the method is correct and the result is incorrect only because the inputs to the goal appropriately came from a previous goal that incorrectly computed those inputs.

In making the above deductions, try to distinguish between errors that can be explained by a single fault and those that are composites of two or more faults. The following example could be conceived as a single error in which the student has mistakenly converted a decimal representation to time. This would constitute a single error for which 1 point would be deducted.

Time1 = 10.67
Time1 = 11 hr 7 min

In contrast, the following production could be interpreted as two separable errors, one in failing to round 10.66 to 10.67 (the result of 800/75), and the second in confusing decimal and time representations. For this goal, one point would be deducted for each of these computational mistakes.

Time1 = 800/75
Time1 = 11 hr 6 min

5. Unless the final answer (the value on the ANSWER line) is redundant with the culminating value in the student's solution, treat this final answer as part of the solution proper. That is, in many student solutions the ANSWER line value is <u>not</u> redundant but instead represents the result of the student's last goal. Such values should be included in scoring that goal.

6. Treat as equivalent the various operational notations (e.g., *, x, (), ·); mixed numbers and improper fractions (e.g., $8^1/_3$ and $^{25}/_3$); numbers with and without units (400 and 400 doses); and percentages, decimals, and fraction equivalents (e.g., $^1/_4$%, .25%, .0025, and $^1/_{400}$).

7. Treat as correct a goal that is satisfied except for the presence of a unit conversion if that conversion is made in a subsequent goal. In the example below, treat equivalently the conversion of hours to hours and minutes whether it occurs in goal #5, goal #4, or goals #1 and #2.

Problem: On a 600-hundred mile motor trip, Bill averaged 45 miles per hour for the first 285 miles and 50 miles per hour for the remainder of the trip. If he started at 7:00 a.m., at what time did he finish the trip (to the nearest minute)?

a. Time 1 = 285 miles / 45 miles per hour
   Time 1 = 6.33 hours (6.33 hours = 6 hours and 20 minutes)
b. Distance 2 = 600 miles - 285 miles
   Distance 2 = 315 miles
c. Time 2 = 315 miles / 50 mile per hour
   Time 2 = 6.3 hours (6.3 hours = 6 hours and 18 minutes)
d. Total time = 6.33 hours + 6.3 hours
   Total time = 6 hours 20 min + 6 hours 18 min
   Total time = 12 hours 38 min
e. End time = 7:00 a.m. + 12 hours 38 min (7:00 a.m. + 12.63 hrs = 7:38 p.m.)
   End time = 7:38 p.m.

8. In some cases, the scoring key for a problem presents two alternative goal decompositions. Score the examinee response according to the decomposition that best characterizes the response. Be sure to use the same maximum scores and the same point deduction rules <u>regardless</u> of the decomposition being used to score the response. Under this rule, partially correct solutions that follow more efficient decompositions will generally receive more points than will similar quality solutions following less efficient decompositions.

9. The minimum score for a goal is 0, as is the minimum total score for a solution.

Appendix C

Canonical Solutions and Bug Definitions
for the Original Three Problem Classes

From Fitting new measurement models to GRE General Test constructed-response item data (RR-91-60) by R. E. Bennett, M. M. Sebrechts, and K. Yamamoto. (Princeton, NJ: Educational Testing Service, 1991).

Canonical Solutions

Below are standard-form solutions. Following these canonical solutions are uniquely named and numbered bugs, their definitions, and examples. Examples are shown as deviations from the canonical solutions. The solution from which the example deviates is indicated by D=RT for distance problems, % for percent problems, and WORK for work problems. In most cases only the relevant modified lines are given. These lines are in most cases numbered and matched to the canonical version.

D=RT

1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6.33 hours
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
5. Time2 = 315 miles / 50 miles per hour
6. Time2 = 6.3 hours
7. Total Time = 6.33 hours + 6.3 hours
8. Total Time = 6 hours 20 min + 6 hours 18 min
9. Total Time = 12 hours 38 min
10. Finish Time = 7:00 a.m. + 12 hours 38 min
11. Finish Time = 7:38 p.m.

Percent (%)

Solution A:

1. 5% = .05
2. Annual Dividend = .05 * $750
3. Annual Dividend = $37.50
4. Investment Time = $750 / $37.50 per year
5. Investment Time = 20 years

Solution B:

1. 5% dividend per year * X years = 100% dividend
2. X years = 100% dividend / 5 % dividend per year
3. X = 20 years

Work

1. Net Filling Rate = 20 cc per minute - 4 cc per minute
2. Net Filling Rate = 16 cc per minute
3. Filling Time = 2000 cc / 16 cc per minute
4. Filling Time = 125 minutes

Math Bugs

Remainder as Decimal (103): The remainder of a division is treated as a decimal. For example, 10/3 = 3.1

D=RT 2. Time1 = 6.15 hours

Close Enough (105): The value is not exact, but is accepted as being within a reasonable margin of error. This is used to catch potential spurious deviations, which are not precision errors (e.g. 8,796 for 8,795).

D=RT 4. Distance2 = 312 miles

Close-Enough Units (106): Obtained and expected values match up to the units place but fail to match in the tenths place.

D=RT 2. Time1 = 6 hours

Close-Enough Tenths (107): Obtained and expected values match up to the tenths place but fail to match in the hundredths place. Close-enough bugs are tested in order. First, close-enough-tenths is tested, then close-enough-units.

D=RT 2. Time1 = 6.30 hours

Decimal Shift (108): The obtained value has a shifted decimal with respect to the expected value.

D=RT 2. Time1 = 63.3 hours

General Plan Bugs

No Reduction (201): An expression is not sufficiently reduced. This bug is reported only if the nonreduced value is not resolved later in the solution.

D=RT 1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6.33 hours
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
5. Time2 = 315 miles / 50 miles per hour
7. Total time = 6.33 hours + 6.3 hours
8. Total time = 6 hours 20 min + 6 hours 18 min

No Final Reduction (202): The "final" answer to the problem is not reduced. It is like the no-reduction bug but applies to the final goal. The difference is largely technical, indicating the way in which the bugs are treated during processing. A no-reduction bug can be subsequently resolved. A no-final-reduction bug cannot.

D-RT 1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6.33 hours
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
5. Time2 = 315 miles / 50 miles per hour
6. Time2 = 6.3 hours
7. Total Time = 6.33 hours + 6.3 hours
8. Total Time = 6 hours 20 min + 6 hours 18 min
9. Total Time = 12 hours 38 min
10. Finish Time = 7:00 a.m. + 12 hours 38 min

Times for Divide (203): The student uses multiplication where division is required.

D-RT 1. Time1 = 285 miles * 45 miles per hour

Divide for Times (204): The student uses division where multiplication is required.

%A 2. Annual Dividend = .05 / $750

Add for Subtract (205): Addition is used where subtraction is required.

D-RT 3. Distance2 = 600 miles + 285 miles

Unknown Value (206): The student uses a value in his or her solution that fits the required structure but that cannot otherwise be accounted for by math or more specific plan errors.

D-RT 1. Time1 = 285 miles * 45 miles per hour
2. Time1 = 5

Out-of-Plan Value (207): A value used in a subsequent goal fits into the overall plan structure, but it is unclear what the source of the value is.

D-RT 7. Total Time = 5.6 hours + 5.3 hours
8. Total Time = 5 hours 36 min + 5 hours 18 min
9. Total Time = 10 hours 54 min
10. Finish Time = 7:00 a.m. + 10 hours 54 min
11. Finish Time = 5:54 p.m.

Unexplained Value (208): A plan has a single unexplained value within the correct structure. This bug is used in INTEREST and WORK problems. D-RT uses more specific bugs, 401 and 402.

WORK 3. Filling time = 2000 cc / 64 cc per minute

Common Symbol (209): This bug is triggered whenever the label is used for identification. It indicates that the combination of other plans and bugs could not adequately explain the value. At the same time, the value assigned to the bug must be within a reasonable range of the expected value, in this case within 1% of the larger of the obtained and expected values.

D=RT 3. absent
4. Distance2 = 312 miles

Specific Plan Bugs

D=RT Time Bugs

Remainder as Time (301): The remainder in a division is treated as a time unit.

D=RT 1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6 hours 15 min

Decimal as Time (302): The decimal portion of a division is treated as a time unit.

D=RT 1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6 hours 33 min

Decimal-Portion as Time (303): The same as decimal-as-time (302) except that it does not assume access to the current match frame.

D=RT 600 m - 285 m = 315 m
285 m / 45 min = 6.33 hr
315 / 50 min = 6.3 hr
6.33 * 45 = 28485
6.33 + 6.30 = 12.63
6.33 + 6.30 = 13.03 hrs
7 + 13.03 = 8:03 p.m.
ANSWER = 8:03 p.m.

Time as Decimal (304): A time unit is treated as a decimal.

D=RT 1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6 hours 20 min
Time1 = 6.20 hours

AM/PM Shift (305): The student changes the time from a.m. to p.m. or p.m. to a.m.. This bug is sometimes triggered by student failure to indicate either a.m. or p.m., in which case a.m. is assumed.

D=RT 11. Finish Time = 7:38 a.m.

Close Enough Minutes (306): The observed and expected values are not exactly the same, but are within 1 minute of each other.

D=RT 2. Time1 = 6.34 hours

Other D-RT Bugs

Unexplained Rate (401): The value for the rate in a structurally matched plan is incorrect.

D-RT 1. Time1 = 285 miles / 55 miles per hour

Unexplained Distance (402): The value for the distance in a structurally matched plan is incorrect.

D-RT 3. Distance2 = 500 miles - 285 miles

Rate2 for Rate1 (404): The second rate is used in place of the first rate.

D-RT 1. Time1 = 285 miles / 50 miles per hour

Rate1 for Rate2 (405): The first rate is used in place of the second rate.

D-RT 5. Time2 = 315 miles / 45 miles per hour

Whole for Part Distance (406): The total distance is used to determine one of the partial times instead of the partial distance.

D-RT 5. Time2 = 600 miles / 50 miles per hour

Part for Whole Time (407): The ending time uses only one of the elapsed times instead of the total elapsed time.

D-RT 10. Finish Time = 7:00 a.m. + 6 hours 20 min

Unknown Time1, Unknown Time2, Unknown Total Time (411, 412, 413): These three "Unknown" bugs are part of a single plan that finds a global solution structure in the absence of reasonable constituent values. If only a single bug occurs or the numbers are reasonable deviations, the solution will usually be matched by a series of individual bugs or by the use of implicit matching. The following example incorporates the three "unknown" bugs (time1, time2, and total time).

D-RT 1. Time1 = 200 miles / 40 miles per hour
2. Time1 = 5 hours
5. Time2 = 300 miles / 60 miles per hour
6. Time2 = 5 hours
7. Total time = 5 hours + 5 hours
9. Total time = 10 hours
10. Finish Time = 7:00 a.m. + 10 hours
11. Finish Time = 5:00 p.m.

Unweighted Average Rate (414): An average rate is computed improperly since it is not weighted according to the different times.

D-RT Average Rate = (15 pages per min + 50 pages per min)/2
Average Rate = 32.5 pages per min
Average Time = 720 pages / 32.5 pages per min
Average Time = 22.15 min

Percent Bugs

Percent as Decimal (501): The percent value is treated as a decimal (e.g., 5% is treated as 5.0).

%A 1. absent
2. Annual Dividend = 5 * $750
3. Annual Dividend = $3750
4. Investment Time = $750 / $3750 per year
5. Investment Time = 0.20 years

Decimal as Percent (502): A decimal value is treated as a percent (e.g., .05 as if it were .05% or .0005).

%A 2. Annual Dividend = 0.05 * $750
3. Annual Dividend = $0.3750

Mixed Percent and Decimal (503): The student mixes decimal and percent values.

%B 1. 0.05 dividend per year * X years = 100% dividend
2. X years = 100% dividend / 0.05 dividend per year
3. X = 2000 years

Result per Unit-Amount/Rate (504): The dividend is calculated as a division of investment by rate instead of a multiplication of investment times the rate. This bug also applies to calculating the profit per load, amount earned per shift, and active ingredient per dose.

%A 2. Annual Dividend = $750 / 0.05

Work Bugs

Subtract Order (603): The items are subtracted in the wrong order (i.e., volume out minus volume in).

WORK 1. Net filling rate = 4 cc per minute - 20 cc per min

Missing Goals

These bugs indicate that the stated goal is missing. No goal can have any other associated bug if it is "missing."

D=RT Bugs

Missing First Goal (911): Time1 is missing. The goal takes the canonical form, Time1 = Distance1 / Rate1.

D=RT
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
5. Time2 = 315 miles / 50 miles per hour
6. Time2 = 6.3 hours
10. Finish Time = 7:00 a.m. + 12 hours 38 min
11. Finish Time = 7:38 p.m.

Missing Second Goal (912): Distance2 is missing. The goal takes the canonical form, Distance2 = Total Distance - Distance1.

D=RT
1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 5.7 hours
5. Time2 = 2.85 miles / 50 miles per hour
6. Time2 = 6.3 hours
7. Total time = 5.7 hours + 6.3 hours
8. Total time = 5 hours 42 min + 6 hours 18 min
9. Total time = 12 hours
10. Finish Time = 7:00 a.m. + 12 hours
11. Finish Time = 7:00 p.m.

Missing Third Goal (913): Time2 is missing. The goal takes the canonical form, Time2 = Distance2 / Rate2.

D=RT
1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6.33 hours
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
10. Finish Time = 7:00 a.m. + 6 hours 20 min
11. Finish Time = 1:20 p.m.

Missing Fourth Goal (914): Total Time is missing. The goal takes the canonical form, Total Time = Time1 + Time2.

D=RT
1. Time1 = 285 miles / 45 miles per hour
2. Time1 = 6.33 hours
3. Distance2 = 600 miles - 285 miles
4. Distance2 = 315 miles
5. Time2 = 315 miles / 50 miles per hour
6. Time2 = 6.3 hours
10. Finish Time = 7:00 a.m. + 6 hours 20 min
11. Finish Time = 1:20 p.m.

Missing Fifth Goal (915): Finish Time is missing. The goal takes the canonical form, Finish Time = Start Time + Total Time.

```
D=RT  1. Time1 = 285 miles / 45 miles per hour
      2. Time1 = 6.33 hours
      3. Distance2 = 600 miles - 285 miles
      4. Distance2 = 315 miles
      5. Time2 = 315 miles / 50 miles per hour
      6. Time2 = 6.3 hours
      7. Total time = 6.33 hours + 6.3 hours
      8. Total time = 6 hours 20 min + 6 hours 18 min
      9. Total time = 12 hours 38 min
```

Percent Bugs

Missing First Goal (921): The Percent Conversion is missing. The goal takes the canonical form, Decimal = .01 * percent

Missing Second Goal (922): Annual Dividend is missing. The goal takes the canonical form, Annual Dividend = Rate Per Year * Investment.

```
%A    4. Investment Time = $750 / 5
      5. Investment Time = 150 years
```

Missing Third Goal (923): Investment Time is missing. The goal takes the canonical form, Investment Time = Investment / Dividends Per Year.

```
%A    1. 5% = .05
      2. Annual Dividend = .05 * $750
      3. Annual Dividend = $37.50
```

Work Bugs

Missing First Goal (931): Net Filling Rate is missing. The goal takes the canonical form, Net Filling Rate = Rate In - Rate Out.

```
WORK  3. Filling time = 2000 cc / 20 cc per minute
      4. Filling time = 100 minutes
```

Missing Second Goal (932): Filling Time is missing. The goal takes the canonical form, Fill Time= Volume / Net Rate.

```
WORK  1. Net filling rate = 20 cc per minute - 4 cc per minute
      2. Net filling rate = 16 cc per minute
```

BEST COPY AVAILABLE

54020-07629 • Y43P.7 • 209091